

國立交通大學

電信工程研究所

碩士論文

分散式籬柵空時碼接收機設計

Receiver Design for Distributed Space Time Trellis Codes

研究生：塗明穎

指導教授：張文鐘 博士

中華民國 九十九 年 七 月

分散式籬柵空時碼接收機設計

Receiver Design for Distributed Space Time Trellis Codes

研究生：塗明穎

Student : Ming-Ying Tu

指導教授：張文鐘 博士

Advisor : Wen-Thong Chang

國立交通大學

電信工程研究所

碩士論文

A Thesis

Submitted to Institute of Communication Engineering
College of Electrical Engineering and Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Communication Engineering

July 2010

Hsinchu, Taiwan, Republic of China

中華民國 九十九 年 七 月

分散式籬柵空時碼接收機設計

研究生：塗明穎

指導教授：張文鐘 博士

國立交通大學電信工程研究所

摘要

無線通訊中，在慢速衰弱通道環境下，多天線系統(MIMO)[1]加上籬柵空時碼(STTC)是有效改善傳輸錯誤率的方案之一。籬柵空時碼會根據輸入資訊與編碼器狀態，乘上行生係數，產生發射天線訊號。手持設備因成本與尺寸考量問題，往往僅有單一天線；傳送者欲使用多重天線的特性來幫助傳輸品質，可藉由合作式通訊達到目的。典型的 DF 合作式通訊，傳送端先將訊號傳給中繼端群，中繼端群將接收訊號解碼過後，再重新編碼傳至接收端。當傳送端與某一中繼端間通道的雜訊干擾甚為嚴重時，中繼端解出的訊號易發生錯誤，此時中繼端假如對此錯誤訊號加以重新編碼，再傳至接收端，會導致接收端解碼錯誤；中繼端群對於不同的訊號作編碼，會導致中繼端群編碼器狀態不一致。本篇論文提出混合式籬柵解碼，來改善此狀態不一致的問題，使得接收端不至於完全喪失解碼能力。當編碼器狀態不一致時，我們稱之為”虛擬狀態”。接收端應用混合式籬柵碼，結合虛擬狀態，改變原有籬柵空時碼結構，以系統觀點來重新定義籬柵的輸出輸入關係；與傳統籬柵空時碼比較，混和籬柵解碼犧牲了錯誤率上的分集程度，來解決中繼端群狀態不一致所引發的解碼問題。此外，吾人利用空時碼概念，討論混合式籬柵碼的分集性能、編碼增益與選擇增益，並在搜尋籬柵空時碼時，提出有效率的演算法。

Receiver Design for Distributed Space Time Trellis Codes

Student: Ming-Ying Tu

Advisor: Wen-Thong Chang

Institute of Communication Engineering

National Chiao Tung University

Abstract

MIMO with STTC is one of the powerful methods to improve error rate performance in slow fading channel. The signal transmitted by STTC is based on the convolution of the input bit plus memory bit with connection coefficient. Due to the size limitation, most mobile and handheld equipment usually has only one antenna. If MIMO is to be applied, cooperative communication would be a good choice. For DF cooperative communication, source terminal transmits signal to relay in phase 1. After relay decoding the signal, it will re-encode the signal and transmit it to destination in phase 2. However, possibility exists that the relay may decode the signal erroneously. In this case, a series of re-encoded error message will be sent to the destination. In this condition, the conventional trellis decoder will not be able to find the correct path and retrieve the correct information bits. One way to solve this problem is to examine the state status of the encoder. When error bit appears, the states in each relay will not be consistent. In this thesis, we expand the conventional trellis state to incorporate all these inconsistent states. We call this as the hybrid STTC decoding. Another simple approach is to estimate the transmitted signal of each relay separately, and each signal is trellis decoded separately. In this way, the diversity advantage of the MIMO is lost. The hybrid STTC which combines virtual state and correct state to consider the inconsistent problem can still have the advantage of the MIMO diversity. But the diversity decreases as more states are incorporated. Compared with the traditional STTC, hybrid STTC trades the diversity for error detection.

誌 謝

時光荏苒，兩年的時光匆匆流逝，碩士生涯至此也算圓滿結束，對於我的指導教授張文鐘博士，心中只有由衷的感謝，老師對於研究的專業，總是能夠在我研究遭遇困難及瓶頸時給予我最精準又最精闢的指導，處事踏實又懂得享受生活的人生哲學，老師給我們研究生做了一個最好的榜樣。同時也感謝口試委員王晉良教授、林銀議教授與林大衛教授，各位口試委員都以精準的眼光和專業能力給了研究上和論文上的建議，讓我的論文能夠更加的完備，很感謝他們。

對於能成為 821 實驗室的一員我滿懷感謝，實驗室的學長、學弟妹和同學們都是我不論研究上或是生活上的好夥伴，博班學長家豪；96 級學長秉謙、建民、盛如、志偉、弘達、如宏、琮壹；98 級和 99 級學弟妹舒評、信妤、維哲、耀駿、詩倩；97 級同學怡如、雅嵐、耀葦；助理立杰。有了大家在，讓我的研究生活精彩非凡，希望大家還能時常相聚。

最後也是最重要的，我最感謝我的家人，家中父母親總不吝惜長途電話費，讓我能聽到他們聲音，了解家鄉近況，一解思鄉之愁；妹妹千儀、千慧像朋友一樣給了我許多建議，增加自己的自信；還有要感謝交大土地公的保佑，讓我可以 在交大兩年都很平順。沒有大家的支持，就沒有現在的我，我很感謝。

誌於 2010.夏 新竹。交大

明穎

目 錄

摘 要	I
ABSTRACT	II
誌 謝	III
目 錄	IV
圖 目 錄	VII
表 目 錄	IX
第一章 緒論	1
1.1 前言	1
1.2 研究動機	1
1.3 研究簡介	2
1.4 論文架構	3
第二章 多重天線系統介紹	4
2.1 多重天線的通道容量 (MIMO SYSTEM CAPACITY DERIVATION)	4
2.2 應用於多重天線的分集技術—空時編碼效能討論	6
2.2.1 類慢速衰減通道下的設計準則	7
2.2.2 類慢速衰減通道下的空時籬柵碼設計準則模擬	8
2.3 多重天線的多工與分集(MIMO DIVERSITY MULTIPLEXING)	9
APPENDIX	11
第三章 空時編碼技術	14
3.1 籬柵空時碼(SPACE TIME TRELLIS CODES)	14
3.1.1 籬柵空時碼架構	14
3.1.2 籬柵空時碼架構特性	15
3.1.3 應用於籬柵解碼的討論	17
3.1.3.1 Viterbi decoder	17
3.1.3.2 M-algorithm	18
3.1.3.3 Adaptive M-algorithm	18
3.1.3.4 BCJR algorithm	18
3.1.4 籬柵空時碼模擬	19
3.1.4.1 籬柵空時碼不同解碼方式比較	19
3.1.4.2 記憶性與籬柵空時碼的關係	20

3.1.4.3 使用 M-Algorithm 於維特比解碼.....	21
3.2 區塊空時碼(SPACE TIME BLOCK CODE)	22
3.3 多層次空時碼(LAYERED SPACE TIME CODE)	23
第四章 空時籬柵碼之搜尋演算法	26
4.1 全域搜尋演算法(EXHAUSTIVE SEARCH)	26
4.2 基於天線個數之遞迴演算法(PRAGMATIC STTC)	27
4.3 應用於空時籬柵碼搜尋之維特比演算法	28
4.4 實際模擬驗證	32
4.5 應用於空時籬柵碼 DETERMINANT CRITERION 搜尋之維特比演算法問題.....	32
4.5.1 應用於空時籬柵碼 Determinant Criterion 搜尋之維特比演算法.....	32
4.5.2 應用於空時籬柵碼 Determinant Criterion 搜尋之維特比演算法問題.....	33
第五章 合作式通訊的應用	34
5.1 COOPERATIVE COMMUNICATION PROTOCOL 簡介.....	34
5.1.1 Amplify and Forward.....	34
5.1.2 Decode and Forward.....	34
5.2 DF 合作式通訊—符碼(SYMBOL)錯誤率分析	35
5.2.1 A simplified Cooperation model	35
5.2.2 Alamouti code in DF cooperation model	36
5.2.3 Alamouti code in DF cooperation model without source destination link.....	38
5.3 DF 合作式通訊—封包錯誤率分析	40
5.3.1 應用於 DF 合作式通訊上的籬柵空時編碼.....	40
5.3.2 分散式籬柵空時碼編碼結構	40
5.3.3 定義分散式籬柵空時碼矩陣	41
5.3.4 接收端的解碼器	42
5.3.5 慢速衰減通道下的設計準則	42
5.3.6 類慢速衰減通道下的設計準則	44
5.3.7 比較慢速與類慢速衰減通道狀況	45
5.4 應用於合作式通訊上的籬柵空時編碼實驗.....	45
5.4.1 Hard/Soft decision 解碼比較.....	45
5.4.2 不同 system model 效能上的比較.....	46
5.4.2.1 A simplified Cooperation model	46
5.4.2.2 DSTTC in DF cooperation model	47
5.4.2.3 DSTTC in DF cooperation model without S-D.....	49
5.4.2.4 比較前面三節的封包錯誤率：	50
第六章 分散式籬柵空時碼接收機設計	53
6.1 系統架構.....	53

6.2 傳統接收端解碼方式所面臨的問題.....	54
6.3 利用具延遲特性的籬柵空時碼來解決問題.....	57
6.3.1 具延遲特性的籬柵空時碼.....	57
6.3.2 接收端的接收方式.....	57
6.4 分離式 VITERBI 解法.....	59
6.5 混合式(HYBRID STATE)籬柵解碼.....	60
6.5.1 混合式籬柵解碼複雜度討論.....	60
6.5.2 混合式籬柵解碼分級程度討論.....	62
6.5.3 混合式籬柵解碼編碼增益討論.....	64
6.5.4 混合式籬柵解碼的多工與分集討論.....	65
6.5.5 以集合方式討論混合式籬柵解碼—多數決增益 (Voting gain).....	65
6.5.6 多數決混合籬柵解碼使用中繼端群編碼器狀態一致下的探討.....	68
6.5.7 多數決混合籬柵解碼 diversity 特性總結.....	69
6.5.8 應用 M-Algorithm 於混合式籬柵解碼.....	70
第七章 結論.....	72
參考文獻.....	73



圖 目 錄

圖 2-1 MIMO 系統分析圖	4
圖 2-2 籬柵空時碼設計準則驗證結果.....	8
圖 2-3 DIVERSITY-MULTIPLEXING TRADE-OFF	10
圖 3-1 籬柵空時碼架構圖	14
圖 3-2 MEMORY 個數與 CODED SYMBOL 間距離的關係.....	15
圖 3-3 具延遲效應的籬柵空時碼.....	16
圖 3-4 VITERBI 演算法示意圖	17
圖 3-5 HARD、SOFT VITERBI 與 BCJR DECODING 解碼表現比較圖.....	19
圖 3-6 MEMORY 多寡與錯誤率表現的關係圖	20
圖 3-7 MEMORY 多寡與錯誤率表現的關係圖	21
圖 3-8 區塊空時碼架構圖	22
圖 3-9 多層次空時碼架構圖	23
圖 4-1 全域搜尋演算法流程圖	26
圖 4-2 DETERMINANT CRITERION 應用於 VITERBI 演算法	33
圖 5-1 AF PROTOCOL 示意圖	34
圖 5-2 DF PROTOCOL 示意圖	34
圖 5-3 A SIMPLIFIED COOPERATION MODEL 示意圖.....	35
圖 5-4 ALAMOUTI CODE IN DF COOPERATION MODEL 示意圖.....	36
圖 5-5 ALAMOUTI CODE IN DF COOPERATION MODEL WITHOUT SOURCE DESTINATION LINK 示意圖	38
圖 5-6 不同合作式通訊 PROTOCOL 在符碼錯誤率的表現比較.....	39
圖 5-7 DF PROTOCOL WITH S-D LINK IN PHASE 2 示意圖.....	40
圖 5-8 DETERMINANT CRITERION 應用於分散式籬柵空時碼中(FOR SLOW-FADING CHANNEL).....	43
圖 5-9 DETERMINANT CRITERION 應用於分散式籬柵空時碼中(FOR QUASI SLOW-FADING CHANNEL)	45
圖 5-10 HARD V.S. SOFT DECISION 比較.....	46

圖 5-11 5.2.1 節增加分散式籬柵空時碼保護後，封包錯誤率表現.....	50
圖 5-12 5.2.2 節增加分散式籬柵空時碼保護後，封包錯誤率表現.....	51
圖 5-13 5.2.3 節增加分散式籬柵空時碼保護後，封包錯誤率表現.....	51
圖 5-14 比較三種傳輸方式的封包錯誤率表現.....	52
圖 6-1 系統存在惡意中繼端的傳送訊號模式.....	54
圖 6-2 分散式系統傳收端的區塊圖.....	54
圖 6-3 受干擾中繼端發射錯誤訊號至接收端，造成傳統籬柵解碼錯誤例子一(有機會成功解碼).....	55
圖 6-4 受干擾中繼端發射錯誤訊號至接收端，造成傳統籬柵解碼錯誤例子二(無解碼成功機會).....	56
圖 6-5 傳統解碼方法在中繼端群編碼器狀態一致與不一致情況下，錯誤率比較圖。 .	56
圖 6-6 使用具延遲特性的籬柵空時碼來解決問題時，所使用的接收端架構.....	57
圖 6-7 經過 LST DETECTOR 後，計算中繼端的方法。	58
圖 6-8 傳送端使用 DELAY DIVERSITY STTC，接收端使用圖 6-6 方式來解決中繼端群狀態不一致問題，其錯誤率表現圖。	58
圖 6-9 分離式維特比解法的架構圖.....	59
圖 6-10 惡意中繼端對於不同的 INFORMATION BITS 編碼的示意圖.....	60
圖 6-11 三個中繼端在不同的 STATE 對應到新的 STATE 示意圖.....	61
圖 6-12 混合式籬柵解碼(ML)示意圖	62
圖 6-13 應用混合式籬柵碼解決中繼端群編碼器狀態不一致問題.....	64
圖 6-14 混合籬柵碼的編碼增益比較.....	65
圖 6-15 混合式籬柵解碼與傳統籬柵空時碼解碼的關係圖.....	66
圖 6-16 選擇增益比較圖.....	67
圖 6-17 具備多數決混合籬柵碼與具備多數決分離式籬柵碼的效能比較.....	67
圖 6-18 中繼端狀態一致下，傳統解碼與混和解碼錯誤率比較圖.....	69
圖 6-19 M-ALGORITHM 應用於混合式籬柵空時碼來降低解碼複雜度.....	70

表 目 錄

表 3-1	UPPER BOUND OF THE RANK VALUES FOR STTC	16
表 3-2	BCJR 參數的計算方法	18
表 4-1	使用 PRAGMATIC+VITERBI 演算法於搜尋碼上之驗證結果.....	32
表 6-1	具延遲特性之籬柵空時碼.....	55



第一章 緒論

1.1 前言

近年來，無線通訊技術被廣泛應用在手持設備、行動電腦、車載服務等，人們對於取得資料比起以往更為方便。然而，對於環境的變化多端、頻譜資源昂貴、顧客要求手持設備價格降低等，使得各方學者、工程師等積極研究多重天線的特性，期望能夠增加通訊的傳輸量或者可靠度，用以增加通訊時的頻譜使用效率及無線通訊的普及化。

空時碼[2]目的是降低通訊系統的傳輸錯誤率，使資料能做更可靠的傳輸，尤其對於衰減通道的影響，更為有效果。單一路徑(single path)的通訊系統中，衰減通道對於訊號造成衝擊，使得接收端估測的能力，無法與 AWGN 通道(不考慮通道衰減)下的估測能力相提並論。這是由於單一路徑上的通道衰減係數，決定解碼端的表現。分集技術(diversity)能夠改善衰弱通道所造成的影響，運用空間、時間或頻率的自由度，使得訊號在這些自由度上重複傳輸，產生多重路徑，假如每條路徑上的通道係數彼此獨立產生，訊號因單一路徑影響而被破壞的機率降低，如此一來，能增加接收端估測訊號的正確率。空時碼的應用，結合多天線與錯誤更正碼，能降低訊號在衰減通道下的傳輸錯誤率，更是實現分集特性的有效方法之一。

同時擁有發射分集、編碼增益、抵抗衰減通道，對於提高頻譜效率使用上，籬柵空時碼能提供解決方案。籬柵空時碼會根據輸入資訊與編碼器狀態，乘上衍生係數，產生發射天線訊號，將資訊能夠散播在空間與時間上，增加通訊的可靠程度。

因成本或尺寸問題，手持裝置通常僅有單一天線，欲使用多重天線的傳輸特性時，應用合作式通訊[3]於其中，可藉由中繼端群的幫助，增加傳輸訊號的可靠程度。合作式通訊中，傳送端先將訊號傳至中繼端群，中繼端群再對其接收訊號作處理，送至接收端作估測。合作式通訊 protocol 大致上分為 AF (Amplify-and-Forward)、DF (decode-and-forward)，其他種的 protocol 也依循這兩種方式做延伸。AF 機制中，中繼端群僅對接收訊號作功率正規化後，隨即送出；DF 機制中，中繼端群對接收訊號做解碼之後，再重新編碼，將訊號送至接收端。

當籬柵空時碼運用在合作式通訊時，稱之為分散式籬柵空時碼 DSTTC(Distributed Space Time Trellis Code)[4]；可將原籬柵空時碼中，某一天線的衍生係數轉移到某一中繼端上；如此一來，原本集中式籬柵空時碼需要在傳送端具備多重天線的限制，可以解除；只要使用者能夠找到適當的中繼端(傳送端與中繼端通道佳)，幫助傳送端傳送訊號即可，並可以享有原本籬柵空時碼所帶來的降低錯誤率好處。

1.2 研究動機

相較於其它種的編碼系統，籬柵空時碼顯得較為複雜，不管是在搜尋效能好的編碼器[5]，或是執行解碼演算法的部分，皆須要花費大量的時間與大量的硬體資源，使得它在實際應用上，沒有如 Alamouti scheme 受歡迎。然而，在傳統上，單一天線傳輸系統，已對籬柵碼廣為討論，因此我們可以參照過去的經驗，運用一些知識，應用於空時籬柵碼，尋找好的編碼方式，藉此幫助此編碼系統，更適合於多天線通訊系統上實際操作。

合作式通訊系統也是近期流行的議題，將編碼應用於此系統上時，稱之為分散式空時碼。集中式空時編碼與分散式空時編碼存在的差異性，是編碼應用在合作式通訊系統上必須先了解的基本觀念。此外，終端與中繼設備以不同方法合作，在效能上的差異性，也是個有趣的議題。吾人將在論文中，比較幾個合作式系統的效能，來探討他們的優劣。

當籬柵空時碼應用在合作式通訊時，因系統電路問題，導致某一傳送天線所根據的編碼器狀態(state)與其他天線不同；或者另一種情況，DF(Decode and Forward)合作式通訊中，傳送端與中繼端間的通道受嚴重干擾，中繼端無法正確偵測傳送端訊號 (information bits)，導致中繼端編碼器對於錯誤的資訊，繼續作編碼；接收端收到的訊號，進入傳統籬柵解碼器時，受干擾中繼端訊號影響，接收端在計算路徑距離時會有偏差；甚至在無 AWGN 情況下，傳統籬柵解碼的籬柵中，根本無法找到一條 zero cost 的路徑，容易造成接收端解碼錯誤。因此我們視此受干擾中繼端所發出的訊號，為另一種很強烈的雜訊，造成符碼間的干擾。受到干擾的中繼端，其所編碼的資訊會與其他正常中繼端不同，故受干擾中繼端的編碼狀態過程轉換也與其他正常狀態的中繼端不同，造成中繼端群編碼時，發生狀態不一致的問題。本篇論文提出此類問題的解決辦法—分離式解碼與混合式籬柵解碼。

本論文的目標：

- 探討空時籬柵碼的特性、搜尋碼與在合作式通訊系統上的應用。
- 空時籬柵碼、空時區塊碼在合作式通訊上不同應用的比較。
- 最後探討合作式系統中，中繼端群狀態不一致造成中繼端群所發出的訊號相互干擾，接收端的解決方法。

1.3 研究簡介

尋找具備效能好的空時碼，傳統上，即是將所有可能的編碼方式，比較它們代價(cost)的大小，使用 trace criterion、determinant criterion，決定編碼的好壞；對於某一個空時碼，傳送端 n_T 根輸出天線所形成的 coded symbol matrix，表示如下：

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_L] = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_L^{n_T} \\ x_1^2 & x_2^2 & \dots & x_L^{n_T} \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{n_T} & x_2^{n_T} & \dots & x_L^{n_T} \end{bmatrix} \quad (\text{Eq. 1-1})$$

令 $\mathbf{B} = \mathbf{X} - \hat{\mathbf{X}}$ 、 $\mathbf{A} = \mathbf{B}\mathbf{B}^H$ ，所有可能的 \mathbf{A} 矩陣，成為一個集合。最小的 determinant of \mathbf{A} ，視為此空時碼的 determinant 價值，我們期望 determinant 價值愈高愈好，稱之為 determinant criterion；最小的 trace of \mathbf{A} ，視為此空時碼的 trace 價值，我們期望 trace 價值愈高愈好，稱之為 trace criterion。在比較的過程中，吾人結合兩種觀念，使得搜尋碼在使用 trace criterion 時，所花費的時間縮短。觀念一：假如擁有 n_T-1 根天線的最佳籬柵空時碼編碼係數，可以將此 n_T-1 根天線的編碼係數繼續沿用，根據此編碼係數，搜尋第 n_T 根天線的編碼係數[6]。觀念二：使用 trellis 的特性[7]，對於任意一個編碼方式中，

兩兩 coded symbol 間所構成 coded symbol matrix 的 trace，等同於 trellis 上任意兩條路徑的距離；因此吾人使用 Viterbi 演算法，盡可能快速的比較兩條路徑間的距離；例如： $[0,0,0,\dots,0]$ 代表全為零的 information bits，稱之為 I_0 ； $[0,1,0,\dots,0]$ 代表除了第二個 bit 不為零以外，其它 bits 皆全為零，稱之為 I_1 ， I_0 與 I_1 的差別只在於第二個 bit 相異； $[0,1,X,\dots,X]$ 代表第一個 bit 為 0，第二個 bit 為 1，其它 bits 為任意值，稱之為 $\{I_2\}$ ；換句話說 $\{I_2\}$ 前兩個 bits 與 I_1 一樣。 I_0 、 I_1 與 $\{I_2\}$ 所產生的 coded symbol matrix 為 M_0 、 M_1 與 $\{M_2\}$ ，假使 M_0 與 M_1 的 trace 為 D ，那麼 M_0 與 $\{M_2\}$ 的距離必定大於等於 D 。因此，假如 M_0 與 M_1 形成 coded symbol matrix 的 trace 已經大於 minimum trace 的話， M_0 與 $\{M_2\}$ 形成 coded symbol matrix 的 trace 也一定大於 minimum trace，在搜尋上，就不需花費時間去計算 M_0 與 $\{M_2\}$ 的距離了。

合作式通訊中，面對中繼端編碼器狀態不一致問題時，我們提出分離式籬柵解碼與混合式籬柵解碼。分離式解碼方式，每根天線的輸出值視為不同編碼器所產生出來的符碼，解碼端先將每根天線上的訊號解碼出來，再將每根天線所偵測出來的訊號分別輸入各自對應的解碼器。解出每根天線所傳的資訊(information bits)，最後經由多數決檢查，當其中兩根天線的資訊完全符合傳送端的傳送資訊，即算解碼正確。此方法為簡單的解決途徑，但會引入量化誤差，容易受訊號間干擾，增加接收端的解碼錯誤率，並且接收端的複雜度較傳統籬柵解碼方法高。混合籬柵碼也將每根天線輸出值視為不同編碼器所產生出來的符碼，但將所有不同編碼器結合起來，看做為另一個大型編碼器。傳統上每根傳送天線所依據的編碼器狀態變換過程，具有一致性，所以接收端因籬柵解碼所使用的暫存器數量與傳送端相同。當系統天線被異常或者合作式通訊傳送端與中繼端的通道受到強烈雜訊影響，被干擾的天線或中繼端之狀態變換過程，會與其他正常的天線或中繼端相異。假使接收端能夠記下整體天線或中繼端群的編碼器狀態變化過程，將正常與受干擾中繼端的編碼器狀態，重新定義成新狀態，並重新建立輸入輸出的籬柵關係，如此一來即能偵測通訊系統具有受干擾中繼端的訊號。混合籬柵中，吾人稱編碼器狀態不一致情況時為”虛擬狀態”，傳統籬柵解碼結合虛擬狀態即成為混合籬柵解碼。本論文所使用的混合籬柵解碼(Hybrid STTC)不限定於使用在某種形態的編碼方式，它至少能偵測一個受干擾中繼端所產生的錯誤，並且更正其錯誤；且運用混合籬柵解碼時，不會引入量化誤差，在效能上會優於分離式解碼方式。然而，混合式籬柵碼仍會增加接收端計算複雜度，接收端因籬柵解碼所使用的暫存器數目，會多於任何一個中繼端或傳送端。

1.4 論文架構

本論文第二章先對 MIMO 系統作簡單的介紹。第三章的部分是對於空時編碼作基本認識。第四章，將過去的解碼經驗應用於搜尋碼中，降低搜尋碼的時間。第五章探討，空時碼於合作式通訊上的應用，探討三種系統架構，在效能上的差異。第六章探討合作式系統中，中繼端編碼器狀態不一致時，接收端的解決辦法。第七章為結論。

第二章 多重天線系統介紹

多重天線系統，能提供通訊系統多的自由度，使得收發端能夠承擔更高的資料速度或者使資料能做更可靠的傳輸。多重天線加上空時碼為目前實際系統對於衰弱通道常用的解決方式，本章簡述多重天線與計算空時碼的基本概念，並說明多重天線系統中，分集與多工的相互消長關係，提供後面章節驗證時的一些基本觀念。

2.1 多重天線的通道容量 (MIMO System Capacity Derivation)

相對於傳統單一天線輸出輸入系統，多重輸出輸入系統能夠負擔更高的傳輸速度；此節中，簡單說明多重輸出輸入使系統的通道容量增加。假設多重輸入輸出的通道，可用 $n_R \times n_T$ 為大小的 \mathbf{H} 矩陣來表示，利用 SVD 將 \mathbf{H} 做分解，表示如下：

$$\mathbf{H} = \mathbf{U}\mathbf{D}\mathbf{V}^H \quad (\text{Eq. 2-1})$$

另外，一般的 MIMO 系統，輸出信號 \mathbf{r} 與輸入信號 \mathbf{x} 的對應關係，表示如下：

$$\mathbf{r} = \mathbf{H}\mathbf{x} + \mathbf{n} = \mathbf{U}\mathbf{D}\mathbf{V}^H\mathbf{x} + \mathbf{n} \quad \mathbf{n}: CN(0,1) \quad (\text{Eq. 2-2})$$

其中 \mathbf{D} 裡的每一個元素，為 $\mathbf{H}\mathbf{H}^H$ 或 $\mathbf{H}^H\mathbf{H}$ 的 eigenvalue 方根值。上述方程式，做矩陣轉換：

$$\mathbf{r}' = \mathbf{D}\mathbf{x}' + \mathbf{n}' \quad (\text{Eq. 2-3})$$

$$\mathbf{r}' = \mathbf{U}^H\mathbf{r} \quad (\text{Eq. 2-4})$$

$$\mathbf{x}' = \mathbf{V}^H\mathbf{x} \quad (\text{Eq. 2-5})$$

$$\mathbf{n}' = \mathbf{U}^H\mathbf{n} \quad (\text{Eq. 2-6})$$

將 \mathbf{D} 內的每個 eigenvalue 值表示為 $\sqrt{\lambda_1}, \sqrt{\lambda_2}, \dots, \sqrt{\lambda_r}$ ， r 代表 \mathbf{D} 的 rank 個數， $r < \min(n_R, n_T)$ 。

把上式展開，表示如下：

$$r'_i = \sqrt{\lambda_i}x'_i + n'_i, \quad i = 1, 2, \dots, r \quad (\text{Eq. 2-7})$$

$$r'_i = n'_i, \quad i = r + 1, r + 2, \dots, \text{Max}(n_R, n_T)$$

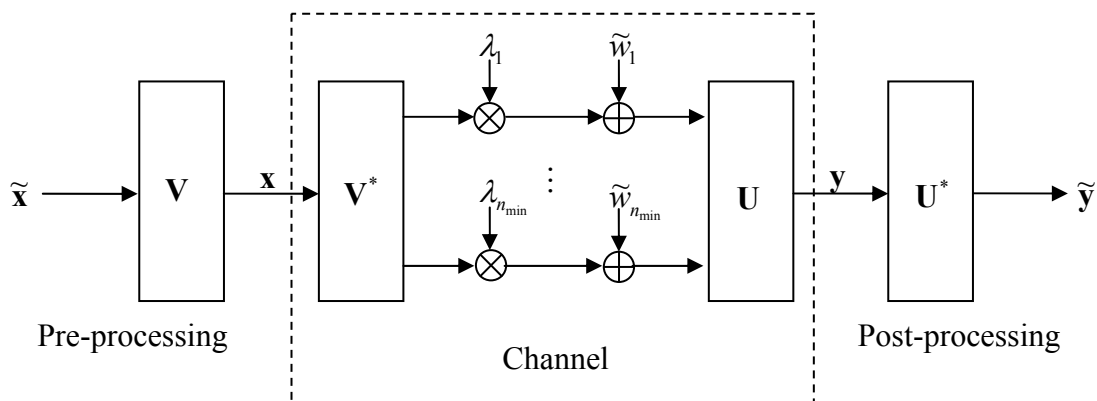


圖 2-1 MIMO 系統分析圖

由上述的關係觀察，接收天線 r'_i ， $i = r + 1, r + 2, \dots, \text{Max}(n_R, n_T)$ ，不受傳送訊號影響。又從式

子 Eq. 2-7，可以把上面的 MIMO 系統，想像成 r 個平行的傳統單一天線輸出輸入系統，如圖 2-1 所示。根據數學定理，假設 \mathbf{U} 為 unitary 矩陣， $\mathbf{U}\mathbf{x}$ 與 \mathbf{x} 有相同的機率分佈：

$$\begin{aligned}\mathbf{R}_{r'r'} &= \mathbf{U}^H \mathbf{R}_{rr} \mathbf{U}、\mathbf{R}_{x'x'} = \mathbf{V}^H \mathbf{R}_{xx} \mathbf{V}、\mathbf{R}_{n'n'} = \mathbf{U}^H \mathbf{R}_{nn} \mathbf{U} \\ \text{tr}(\mathbf{R}_{r'r'}) &= \text{tr}(\mathbf{R}_{rr})、\text{tr}(\mathbf{R}_{x'x'}) = \text{tr}(\mathbf{R}_{xx})、\text{tr}(\mathbf{R}_{n'n'}) = \text{tr}(\mathbf{R}_{nn})\end{aligned}$$

\mathbf{R}_{rr} 代表 r 的 auto-correlation，所以， r' 、 x' 、 n' 與 r 、 x 、 n 有相同的總能量，假設每一根傳送天線是等能量 P/n_T ，傳送訊號至接收端的能量為：

$$P_{ri} = \frac{\lambda_i P}{n_T} \quad (\text{Eq. 2-8})$$

則 MIMO 系統的通道容量可以等效為 r 個平行的傳統單一天線輸出輸入系統，表示如下：

$$C = W \sum_{i=1}^r \log_2 \left(1 + \frac{P_{ri}}{\sigma^2} \right) = W \sum_{i=1}^r \log_2 \left(1 + \frac{\lambda_i P}{n_T \sigma^2} \right) = W \log_2 \prod_{i=1}^r \left(1 + \frac{\lambda_i P}{n_T \sigma^2} \right) \quad (\text{Eq. 2-9})$$

W : bandwidth

$$\text{定義 } \mathbf{Q} = \begin{cases} \mathbf{H}\mathbf{H}^H & n_R < n_T \\ \mathbf{H}^H \mathbf{H} & n_R \geq n_T \end{cases}$$

求 \mathbf{Q} 的 eigenvalues 時，假設 \mathbf{Q} 為 nonsingular matrix， $\mathbf{Q}\mathbf{y} = \lambda\mathbf{y} \rightarrow (\lambda\mathbf{I} - \mathbf{Q})\mathbf{y} = \mathbf{0}$ 。同理 $\det(\lambda\mathbf{I} - \mathbf{Q}) = 0$ 。另外，定義

$$p(\lambda) = \det(\lambda\mathbf{I} - \mathbf{Q}) \quad (\text{Eq. 2-10})$$

因為每個 eigenvalue 的值，皆為 $p(\lambda) = 0$ 方程式的根，所以：

$$p(\lambda) = \prod_i (\lambda - \lambda_i) \quad (\text{Eq. 2-11})$$

將上述(Eq.2-11)與(Eq. 2-12)做結合

$$\prod_i (\lambda - \lambda_i) = \det(\lambda\mathbf{I} - \mathbf{Q}) \quad (\text{Eq. 2-12})$$

將 $\lambda = -\frac{n_T \sigma^2}{P}$ 帶入上式，以得到我們想要的結果：

$$\prod_i \left(1 + \frac{\lambda_i P}{n_T \sigma^2} \right) = \det \left(\mathbf{I} + \frac{P}{n_T \sigma^2} \mathbf{Q} \right) \quad (\text{Eq. 2-13})$$

再帶回(Eq 2-9) 得到

$$C = W \log_2 \det \left(\mathbf{I} + \frac{P}{n_T \sigma^2} \mathbf{Q} \right) \quad (\text{Eq. 2-14})$$

因此，理論證明，MIMO channel 能夠支援更高的通道容量。上述的推導，可以延伸解釋 MIMO 系統，有機會提供更快的傳輸速度。

2.2 應用於多重天線的分集技術—空時編碼效能討論

(MIMO diversity topic- Space time code Performance)

對於多根傳輸天線的傳輸，直覺上我們可以利用它解決兩個問題，

- 將 information 的資訊，分散到每根傳輸天線上做通訊，增加編碼的"冗餘"程度，使得 information 資訊受到多一層保護 → Diversity。
- 每根天線分別傳不同的 information 資訊，增加資料的傳輸速度 → Multiplexing。

相較於上個章節，此章節強調空時碼運用在 MIMO 系統上，錯誤率的表現。根據[2][3]，尋找在瑞雷(Rayleigh)衰減通道下，具備效能好的空時碼，假設 STC 的 frame size 長度為 L ，MIMO 系統擁有 n_T 根傳送天線， n_R 根接收天線($n_T \times n_R$ MIMO 系統)，傳送訊號矩陣、通道矩陣為：

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_t, \dots, \mathbf{x}_L] = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_t^1 & \dots & x_L^1 \\ x_1^2 & x_2^2 & \dots & x_t^2 & \dots & x_L^2 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_1^{n_T} & x_2^{n_T} & \dots & x_t^{n_T} & \dots & x_L^{n_T} \end{bmatrix}_{n_T \times L}$$

$$\mathbf{H}_t = \begin{bmatrix} h_{1,1}^t & h_{1,2}^t & \dots & h_{1,n_T}^t \\ h_{2,1}^t & h_{2,2}^t & \dots & h_{2,n_T}^t \\ \vdots & \vdots & \ddots & \vdots \\ h_{n_R,1}^t & h_{n_R,2}^t & \dots & h_{n_R,n_T}^t \end{bmatrix}_{n_R \times n_T}$$

接收端的訊號、與接收端的 AWGN 雜訊，表示如下：

$$\mathbf{r}_t = [r_t^1, r_t^2, \dots, r_t^{n_R}]^T \quad \mathbf{n}_t = [n_t^1, n_t^2, \dots, n_t^{n_R}]^T \quad \mathbf{r}_t = \mathbf{H}_t \mathbf{x}_t + \mathbf{n}_t \quad (\text{Eq. 2-15})$$

在接收端，ML decision metric 計算如下：

$$\sum_{i=1}^L \sum_{j=1}^{n_R} \left| r_t^j - \sum_{i=1}^{n_T} h_{j,i}^t x_t^i \right|^2 \quad (\text{Eq. 2-16})$$

AWGN，information bit $\{0,1\}$ 為 equally probability 下， $P(\mathbf{X}, \hat{\mathbf{X}})$ ，代表接收端誤認 symbols \mathbf{X} 為 $\hat{\mathbf{X}}$ 的機率可以表示如下：

$$P(\mathbf{X}, \hat{\mathbf{X}} | \mathbf{H}) = Q \left(\sqrt{\frac{E_s}{2N_0}} d_h^2(\mathbf{X}, \hat{\mathbf{X}}) \right) \quad (\text{Eq. 2-17})$$

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2} dt \quad , \quad Q(x) \leq \frac{1}{2} e^{-x^2/2} \quad (x \geq 0)$$

其中 $d_h(\mathbf{X}, \hat{\mathbf{X}})$ 為 symbol 經過通道的影響過後，在星狀圖上的 Euclidean distance，表示如下：

$$d_h^2(\mathbf{X}, \hat{\mathbf{X}}) = \left\| \mathbf{H} \cdot (\hat{\mathbf{X}} - \mathbf{X}) \right\|^2 = \sum_{t=1}^L \sum_{j=1}^{n_R} \left| \sum_{i=1}^{n_T} h_{j,i}^t \hat{x}_t^i - \sum_{i=1}^{n_T} h_{j,i}^t x_t^i \right|^2 \quad (\text{Eq. 2-18})$$

結合上述式子，將 FER 表示如下：

$$P(\mathbf{X}, \hat{\mathbf{X}} | \mathbf{H}) \leq \frac{1}{2} \exp\left(-d_h^2(\mathbf{X}, \hat{\mathbf{X}}) \frac{E_s}{N_0}\right) \quad (\text{Eq. 2-19})$$

2.2.1 類慢速衰減通道下的設計準則

通道環境為類慢速衰減通道(quasi slow-fading channel)時，在一個 framesize 內，通道的值為固定，假設數值由 Rayleigh distribution 產生，然而 frame size 間的通道係數彼此不影響，視為獨立。

$$h_{j,i}^1 = h_{j,i}^2 = \dots = h_{j,i}^L = h_{j,i}; \quad i = 1, 2, \dots, n_T, \quad j = 1, 2, \dots, n_R \quad \{h_{j,i}\} = CN(0,1)$$

類慢速衰減通道下的設計準則，即是將所有可能的編碼方式，比較它們代價(cost)的大小，使用 determinant、trace 準則，決定編碼增益的好壞；對於某一個空時碼，即選定一組編碼方式，傳送端 n_T 根輸出天線所形成的 coded symbol 矩陣，frame size 為 L ，表示如下：

$$\mathbf{X} = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_L^1 \\ x_1^2 & x_2^2 & \dots & x_L^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{n_T} & x_2^{n_T} & \dots & x_L^{n_T} \end{bmatrix}$$

令 $\mathbf{B} = \mathbf{X} \cdot \hat{\mathbf{X}}^{-1}$ 、 $\mathbf{A} = \mathbf{B} \mathbf{B}^H$ ， γ 代表 \mathbf{A} 的秩(rank)， $\{\lambda_i\}$ 代表 \mathbf{A} 的特徵值(eigenvalue)集合。在高訊雜比(SNR)的情況下， $rn_R < 4$ 時，空時碼的封包錯誤率(FER)上界為

$$P(\mathbf{X}, \hat{\mathbf{X}}) \leq \left(\prod_{i=1}^r \lambda_i \right)^{-n_R} \left(\frac{E_s}{4N_0} \right)^{-rn_R} \quad (\text{Eq. 2-20})$$

將空時碼產生的 coded symbol 中，所有可能的 \mathbf{A} 矩陣，成為一個集合 $\{\mathbf{A}\}$ 。最小的 determinant of $\{\mathbf{A}\}$ ，視為此空時碼的 determinant 價值，我們期望 determinant 價值愈高愈好，稱之為 determinant 準則；最小的 rank of \mathbf{A} 視為此空時碼的 rank 價值，rank 代表空時碼的分集程度，我們期望 rank 價值愈高愈好。在高訊雜比(SNR)的情況下， $rn_R \geq 4$ 時，空時碼的封包錯誤率(FER)上界為

$$P(\mathbf{X}, \hat{\mathbf{X}}) \leq \frac{1}{2} \exp\left(-n_R \frac{E_s}{N_0} \sum_{i=1}^{\gamma} \lambda_i\right) \quad (\text{Eq. 2-21})$$

最小的 trace of $\{\mathbf{A}\}$ ，視為此空時碼的 trace 價值，在天線數多的多重天線系統下，我們期望 trace 價值愈高愈好，稱之為 trace 準則。詳細推導過程，可以參照第二章的 Appendix。

- 在 high SNR 狀況、類慢速衰減通道情況下，空時碼的設計準則：
 - $rn_R < 4$ (Low-diversity) → rank criterion、determinant criterion
 - $rn_R \geq 4$ (High-diversity) → trace criterion

2.2.2 類慢速衰減通道下的空時籬柵碼設計準則模擬

根據三個 code，各自擁有不同的 determinant 與 trace 的值，模擬它們彼此間效能上的差異。使用 Visual C++ 軟體，實驗次數 $1e7$ 次，每次實驗皆對編碼器輸入一個封包，每個封包內擁有 260 bits，當一個 bit 出錯時，即記 frame 錯誤一次，調變採用 QPSK，模擬結果如下，

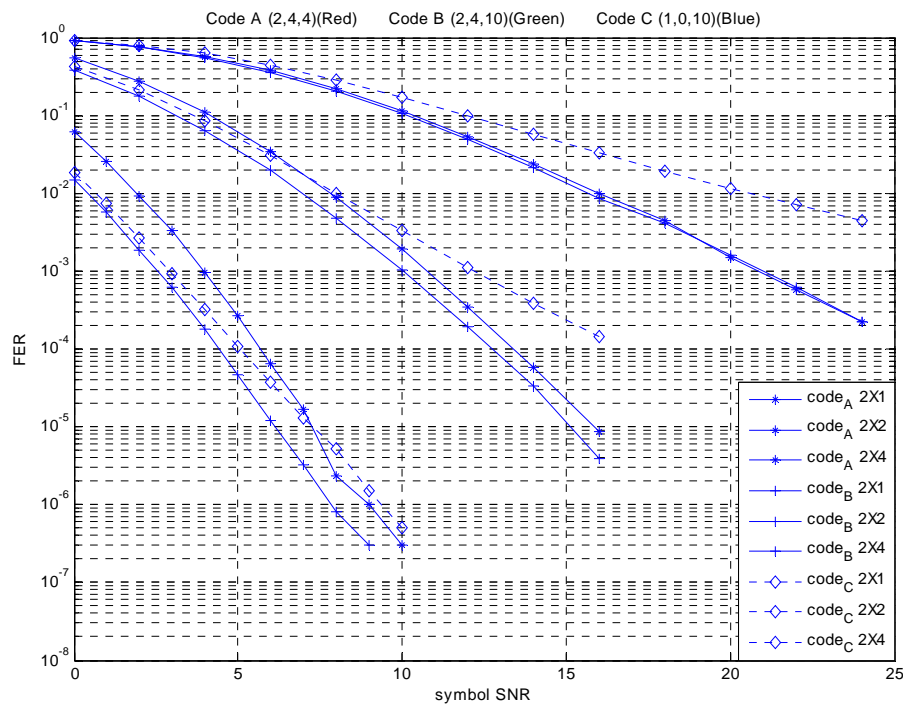


圖 2-2 籬柵空時碼設計準則驗證結果

Code	Generator sequence (Source)	rank	determinant	trace
Code A	$g^1=[(23),(02)]$ $g^2=[(33),(03)]$	2	4	4
Code B	$g^1=[(22),(20)]$ $g^2=[(33),(32)]$	2	4	10
Code C	$g^1=[(32),(02)]$ $g^2=[(23),(23)]$	1	10	10

由圖 2-2 觀察可以發現：

2X1($rn_R < 4$) 時，code A 與 code B 有較高的 rank 與 det，表現的比 code C 好。

2X4($rn_R \geq 4$) 時，code B 與 code C 有較高的 trace，表現的比 code A 好。

經由模擬驗證，rank、determinant、trace criterion 確實正確。也藉由這樣的方法，讓我們找到籬柵空時碼最佳的編碼器，使得在 code modulation 如此複雜的情況下，得到製造 optimal code 的規則。

2.3 多重天線的多工與分集(MIMO diversity multiplexing)

由前面兩小節的推導，不難發現 MIMO 系統不僅能提高通訊容量，也能增加對資訊做有效保護。在此節中，我們討論這兩者的關係，引導出著名的 diversity and multiplexing trade-off。前幾節的推導，都是以 pair-wise error probability 做為切入點，這種推導過程，有一個麻煩之處，就是要對 channel 做大量的積分，才能看到錯誤率與 SNR 的關係；在此章節中，採用另一種方法，outage probability，此方法不僅可以不用考慮 channel 的 distribution，也能觀察 error probability 與 data rate 的關係。

在 SNR 趨近無窮大時，傳輸速度可以用下面式子表示

$$R = r \log SNR \quad (\text{Eq. 2-22})$$

定義 multiplexing gain 為 r ，diversity $\rightarrow d^*(r)$ 如下

$$\lim_{SNR \rightarrow \infty} \frac{\log p_{out}(r \log SNR)}{\log SNR} = -d^*(r) \quad (\text{Eq. 2-23})$$

而 outage probability 中，data rate 與 diversity 的關聯由下方式子表示：

$$p_{out}(R) \approx SNR^{-d^*(r)} \quad (\text{Eq. 2-24})$$

更精確的來說，所謂的 diversity 代表，通訊系統在高 SNR 下，錯誤率的下降速度級數。Multiplexing 代表在高 SNR 下，通訊系統所能提供的多工程度。最佳化的 diversity-multiplexing tradeoff 可以由下列的式子導證：

$$p_{out}^{iid}(r \log SNR) = P \left\{ \log \det \left(\mathbf{I} + \frac{SNR}{n_T} \mathbf{H} \mathbf{H}^* \right) < r \log SNR \right\} \quad (\text{Eq. 2-25})$$

上述式子的物理意義，是根據 Shannon 提出的 capacity 所提出的概念而產生的，當所要傳的資料速度大於通訊系統能負擔的最大通道容量時，傳輸就會發生錯誤，錯誤的機率，即稱之為 outage probability。根據[10]的推導過程，diversity 與 multiplexing 可以獲得以下的關係：

$$(multiplexing, diversity) = (r, (n_T - r)(n_R - r)), r = 0, \dots, n_{min} \quad (\text{Eq. 2-26})$$

以五根傳輸天線與五根接收天線為例子，觀察 diversity 與 multiplexing 的消長關係圖：

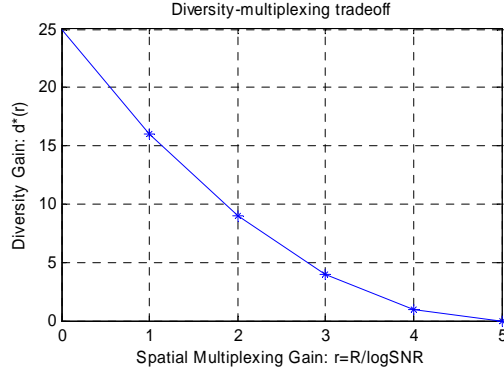


圖 2-3 diversity-multiplexing trade-off

由上圖可知，對於一個 MIMO 系統，必須在 multiplexing 與 diversity 上，做取捨。傳統點對點傳輸，訊號發生錯誤的"事件"表示如下：

$$\log_2(1 + |h|^2 SNR) < R \rightarrow |h|^2 < \frac{2^R - 1}{SNR} \quad (\text{Eq. 2-27})$$

當 R (data rate) 愈高、 SNR 愈小時，錯誤事件就愈可能發生；對於任意 distribution 來說，擁有下列的現象

$$P\{|h|^2 < \varepsilon\} \approx \varepsilon \quad \text{as } \varepsilon \rightarrow 0 \quad (\text{Eq. 2-28})$$

將此現象，引入訊號發生錯誤率中，獲得下列結果

$$P\left\{|h|^2 < \frac{2^R - 1}{SNR}\right\} \approx \frac{2^R - 1}{SNR} \quad \text{as } SNR \rightarrow \infty \text{ or } R \rightarrow 0 \quad (\text{Eq. 2-29})$$

因此，對於一個通道的衰減，在 SNR 很高的狀況下，讓此通訊系統產生錯誤的機會，與 SNR 成反比。將此觀念延伸到 MIMO 系統：

$$P_{out}^{iid}(r \log SNR) = P\left\{\log \det\left(\mathbf{I} + \frac{SNR}{n_T} \mathbf{H}\mathbf{H}^*\right) < r \log SNR\right\} = P\left\{\sum_{i=1}^{n_{\min}} \log\left(1 + \frac{SNR}{n_T} \lambda_i^2\right) < r \log SNR\right\} \quad (\text{Eq. 2-30})$$

因為 MIMO 系統，存在 $n_T \times n_R$ 條路徑 ($\sum_i \lambda_i^2 = \sum_{i,j} |h_{i,j}|^2$)，所以要讓全部路徑的通道增益落於 outage event 內，機率為 $1/SNR^{n_T n_R}$ ，這也就是為什麼，MIMO 通訊系統，較能夠抵抗 fading channel 的原因，當利用到的通道愈多時，傳輸訊號被"壞"通道的影響就能降的愈低。我們必須注意一點，MIMO 的最大 diversity 為 $n_T n_R$ ，是從 Shannon 所提的 capacity 觀點所導證出來的；Shannon 談到：只要資料傳送速率小於通道容量，則必定存在一種 code，達到錯誤率無限小的通訊方式。而從上一節中，設計任何一種空時碼都只是為了達到 Shannon bound 的一種有系統的方式，所以空時碼沒有達到最大 diversity $n_T n_R$ ，只是其中一種的特殊狀況。

傳統討論 diversity 的方式，定義在固定傳輸速度下，錯誤率與訊雜比的關係；在討

論 multiplexing-diversity trade-off 時，通訊速度卻與訊雜比存在關係，

$$R=r \log SNR$$

因此我們可以了解， $r=0$ 不代表通訊時的傳輸速度為零，而是代表傳輸速度固定在一個速率下。因此傳統 diversity 討論方式，要將 r 視為零，帶入 multiplexing-diversity 公式中討論。另外同理，diversity=0 不代表錯誤率為 1，而是代表錯誤率不隨著訊雜比增加而增加。

Appendix

定義 \mathbf{X} 與 $\hat{\mathbf{X}}$ 的 difference 矩陣如下， $[\]^H$ 代表 Hermitian(transpose conjugate)

$$\mathbf{B}(\mathbf{X}, \hat{\mathbf{X}}) = \mathbf{X} - \hat{\mathbf{X}} = \begin{bmatrix} x_1^1 - \hat{x}_1^1 & x_2^1 - \hat{x}_2^1 & \cdots & x_L^1 - \hat{x}_L^1 \\ x_1^2 - \hat{x}_1^2 & x_2^2 - \hat{x}_2^2 & \cdots & x_L^2 - \hat{x}_L^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{n_r} - \hat{x}_1^{n_r} & x_2^{n_r} - \hat{x}_2^{n_r} & \cdots & x_L^{n_r} - \hat{x}_L^{n_r} \end{bmatrix}_{n_r \times L}$$

$$\mathbf{A}(\mathbf{X}, \hat{\mathbf{X}}) = \mathbf{B}(\mathbf{X}, \hat{\mathbf{X}}) \cdot \mathbf{B}^H(\mathbf{X}, \hat{\mathbf{X}}) \quad (\text{Eq. 2-31})$$

因為 $\mathbf{A}(\mathbf{X}, \hat{\mathbf{X}}) = \mathbf{A}^H(\mathbf{X}, \hat{\mathbf{X}})$ ，所以 $\mathbf{A}(\mathbf{X}, \hat{\mathbf{X}})$ 是 nonnegative definite，所有的 eigenvalue 皆 nonnegative。將 $\mathbf{A}(\mathbf{X}, \hat{\mathbf{X}})$ 對角化，並做一些處理得到： $\mathbf{A}(\mathbf{X}, \hat{\mathbf{X}}) = \mathbf{V}^H \mathbf{D} \mathbf{V}$ ； $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{n_r}\}$ 為 $\mathbf{A}(\mathbf{X}, \hat{\mathbf{X}})$ 的 eigenvector， $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{n_r}$ 皆相互正交。 $\mathbf{D} = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_{n_r}\}$ 為 $\mathbf{A}(\mathbf{X}, \hat{\mathbf{X}})$ 的 eigenvalue。假設 $\mathbf{h}_j = [h_{j,1}, h_{j,2}, \dots, h_{j,n_r}]$ ，transmitted symbol 經過 channel 的影響過後，在 constellation 上的 Euclidean distance 重新表示如下：

$$\begin{aligned} d_h^2(\mathbf{X}, \hat{\mathbf{X}}) &= \|\mathbf{H} \cdot (\hat{\mathbf{X}} - \mathbf{X})\|^2 = \mathbf{H} \cdot (\hat{\mathbf{X}} - \mathbf{X}) \cdot (\hat{\mathbf{X}} - \mathbf{X})^H \cdot \mathbf{H}^H = \mathbf{H} \cdot \mathbf{A}(\mathbf{X}, \hat{\mathbf{X}}) \cdot \mathbf{H}^H = \sum_{j=1}^{n_r} \mathbf{h}_j \mathbf{A}(\mathbf{X}, \hat{\mathbf{X}}) \mathbf{h}_j^H \\ &= \sum_{j=1}^{n_r} \mathbf{h}_j \mathbf{V}^H \mathbf{D} \mathbf{V} \mathbf{h}_j^H = \sum_{j=1}^{n_r} \sum_{i=1}^{n_r} \lambda_i |\beta_{j,i}|^2 \quad \text{where } \beta_{j,i} = \mathbf{h}_j \cdot \mathbf{v}_i \end{aligned} \quad (\text{Eq. 2-32})$$

帶入 FER 表示：

$$P(\mathbf{X}, \hat{\mathbf{X}} | \mathbf{H}) \leq \frac{1}{2} \exp\left(-\sum_{j=1}^{n_r} \sum_{i=1}^{n_r} \lambda_i |\beta_{j,i}|^2 \frac{E_S}{N_0}\right) \quad (\text{Eq. 2-33})$$

在機率上，假如 $|h_{i,j}|$ 為 Rayleigh distribution 的話， $|h_{i,j}|^2$ 即為 exponential distribution。此外， \mathbf{v}_i 在 STC 的 generator sequence 決定時也跟著決定了， \mathbf{h}_i 為 random variable vector；所以

$|\beta_{i,j}|$ 的分佈也為 Rayleigh distribution，

$$p(|\beta_{i,j}|) = 2|\beta_{i,j}| \exp(-|\beta_{i,j}|^2) \quad (\text{Eq. 2-34})$$

假設 $\mathbf{A}(\mathbf{X}, \hat{\mathbf{X}})$ 的 rank 為 r ，根據經驗法則，當 $r n_R < 4$ 時，我們只可乖乖對通道積分，求出錯誤率。

$$\begin{aligned} P(\mathbf{X}, \hat{\mathbf{X}}) &\leq \int \cdots \int_{|\beta_{j,i}|=0}^{\infty} \frac{1}{2} \exp\left(-\sum_{j=1}^{n_R} \sum_{i=1}^{n_T} \lambda_i |\beta_{j,i}|^2 \frac{E_S}{N_0}\right) p(|\beta_{1,1}|) p(|\beta_{1,2}|) \cdots p(|\beta_{n_R, n_T}|) \cdot d|\beta_{1,1}| d|\beta_{1,2}| \cdots d|\beta_{n_R, n_T}| \\ &= \left(\prod_{i=1}^r \frac{1}{1 + \frac{E_S}{N_0} \lambda_i} \right)^{n_R} \quad \text{as } \frac{E_S}{N_0} \rightarrow \infty \quad \left(\prod_{i=1}^r \frac{1}{1 + \frac{E_S}{N_0} \lambda_i} \right)^{n_R} \approx \left(\prod_{i=1}^r \lambda_i \right)^{-n_R} \left(\frac{E_S}{4N_0} \right)^{-rn_R} \end{aligned} \quad (\text{Eq. 2-35})$$

由上式可知， $\mathbf{A}(\mathbf{X}, \hat{\mathbf{X}})$ 的 rank、product of eigenvalue 愈大愈好，錯誤率的 upper bound 可以降愈低。又根據經驗法則，當 $rn_R \geq 4$ 時，我們可以適用中央極限定理，對下列式子，求機率分布：

$$D = \sum_{j=1}^{n_R} \sum_{i=1}^{n_T} \lambda_i |\beta_{j,i}|^2 \quad (\text{Eq. 2-36})$$

先證明 $|\beta_{j,i}|$ 為 Rayleigh distribution derived from $CN(0,1)$ ，則 $|\beta_{j,i}|^2$ 為 exponential distribution：

$$P(|\beta_{j,i}|^2 \geq t) = P(|\beta_{j,i}| \geq \sqrt{t}) = \int_{\sqrt{t}}^{\infty} 2|\beta_{j,i}| \exp(-|\beta_{j,i}|^2) = -\exp(-|\beta_{j,i}|^2) \Big|_{|\beta_{j,i}|=\sqrt{t}}^{|\beta_{j,i}|=\infty} = \exp(-t) \quad (\text{Eq. 2-37})$$

$$\therefore P(|\beta_{j,i}|^2 < t) = 1 - \exp(-t)$$

$$\therefore P(|\beta_{j,i}|^2) = \frac{dP(|\beta_{j,i}|^2 < t)}{dt} = \frac{d(1 - \exp(-t))}{dt} = \exp(-t) \rightarrow \text{exponential distribution}$$

所以 $|\beta_{j,i}|^2$ 為 mean=1，var=1 的 exponential distribution。

回來看看我們的 $D = \sum_{j=1}^{n_R} \sum_{i=1}^{n_T} \lambda_i |\beta_{j,i}|^2$ ，注意 $|\beta_{j,i}|^2$ 彼此為相互獨立(存在機率上某些特性，

如 var 可以直接相加或機率可以直接相乘...)

$$D_{mean} = E\left[\sum_{j=1}^{n_R} \sum_{i=1}^{n_T} \lambda_i |\beta_{j,i}|^2\right] = \sum_{j=1}^{n_R} \sum_{i=1}^{n_T} \lambda_i E[|\beta_{j,i}|^2] = \sum_{j=1}^{n_R} \sum_{i=1}^{n_T} \lambda_i = n_R \sum_{i=1}^r \lambda_i \quad (\text{Eq. 2-38})$$

$$D_{\text{Var}} = \text{Var} \left(\sum_{j=1}^{n_R} \sum_{i=1}^{n_T} \lambda_i |\beta_{j,i}|^2 \right) = \sum_{j=1}^{n_R} \sum_{i=1}^{n_T} \lambda_i^2 \text{Var} \left[|\beta_{j,i}|^2 \right] = \sum_{j=1}^{n_R} \sum_{i=1}^{n_T} \lambda_i^2 = n_R \sum_{i=1}^r \lambda_i^2 \quad (\text{Eq. 2-39})$$

經過中央極限定理的估測， D 的機率把它當成 Gaussian random variable with mean= D_{mean} and variance= D_{var} 。 D 的 Gaussian random distribution 如下

$$P(D) = \frac{1}{\sqrt{2\pi D_{\text{var}}}} \exp \left(-\frac{(D-D_{\text{mean}})^2}{2D_{\text{var}}} \right) \quad (\text{Eq. 2-40})$$

如此一來，我們就可以計算 $P(\mathbf{X}, \hat{\mathbf{X}})$ ，結果如下：

$$\begin{aligned} P(X, \hat{X}) &\leq \int_{D=0}^{+\infty} \frac{1}{2} \exp \left(-D \frac{E_S}{N_0} \right) p(D) dD = \frac{1}{2} \exp \left(\frac{1}{2} \left(\frac{E_S}{4N_0} \right)^2 n_R \sum_{i=1}^r \lambda_i^2 - \frac{E_S}{4N_0} n_R \sum_{i=1}^r \lambda_i \right) \\ &\cdot Q \left(\frac{E_S}{4N_0} \sqrt{n_R \sum_{i=1}^r \lambda_i^2} - \frac{\sqrt{n_R \sum_{i=1}^r \lambda_i}}{\sqrt{\sum_{i=1}^r \lambda_i^2}} \right) \quad \because Q(x) \leq \frac{1}{2} \exp \left(-\frac{x^2}{2} \right), x > 0 \quad \text{continue} \rightarrow \\ P(X, \hat{X}) &\leq \int_{D=0}^{+\infty} \frac{1}{2} \exp \left(-D \frac{E_S}{N_0} \right) p(D) dD \leq \frac{1}{2} \exp \left(\frac{1}{2} \left(\frac{E_S}{4N_0} \right)^2 n_R \sum_{i=1}^r \lambda_i^2 - \frac{E_S}{4N_0} n_R \sum_{i=1}^r \lambda_i \right) \\ &\cdot \frac{1}{2} \exp \left(-\left(\frac{E_S}{4N_0} \sqrt{n_R \sum_{i=1}^r \lambda_i^2} - \frac{\sqrt{n_R \sum_{i=1}^r \lambda_i}}{\sqrt{\sum_{i=1}^r \lambda_i^2}} \right)^2 / 2 \right) \\ &= \frac{1}{4} \exp \left(\frac{1}{2} \left(\frac{E_S}{4N_0} \right)^2 n_R \sum_{i=1}^r \lambda_i^2 - \frac{E_S}{4N_0} n_R \sum_{i=1}^r \lambda_i - \left(\frac{E_S}{4N_0} \sqrt{n_R \sum_{i=1}^r \lambda_i^2} - \frac{\sqrt{n_R \sum_{i=1}^r \lambda_i}}{\sqrt{\sum_{i=1}^r \lambda_i^2}} \right)^2 / 2 \right) \approx \frac{1}{4} \exp \left(-n_R \frac{E_S}{4N_0} \sum_{i=1}^r \lambda_i \right) \end{aligned} \quad (\text{Eq. 2-41})$$

由上式可知， $\mathbf{A}(\mathbf{X}, \hat{\mathbf{X}})$ 的 sum of eigenvalue 愈大愈好，錯誤率的 upper bound 可以降愈低。

第三章 空時編碼技術

關於通訊系統達到 Shannon's bound 的討論，Shannon 只假設於 random code、封包長度為無限長，所得到的錯誤率下限；然而，是否真正存在”有系統的”編碼方式，能夠達到 Shannon's bound，各方工程師正在積極探索中，近年來多種空時碼已被提出，在此做一個簡單的介紹，其中籬柵空時碼為本論文的探討重點。

3.1 籬柵空時碼(Space time trellis codes)

3.1.1 籬柵空時碼架構

一般籬柵空時碼編碼器可以用下面的 block 圖來代表。

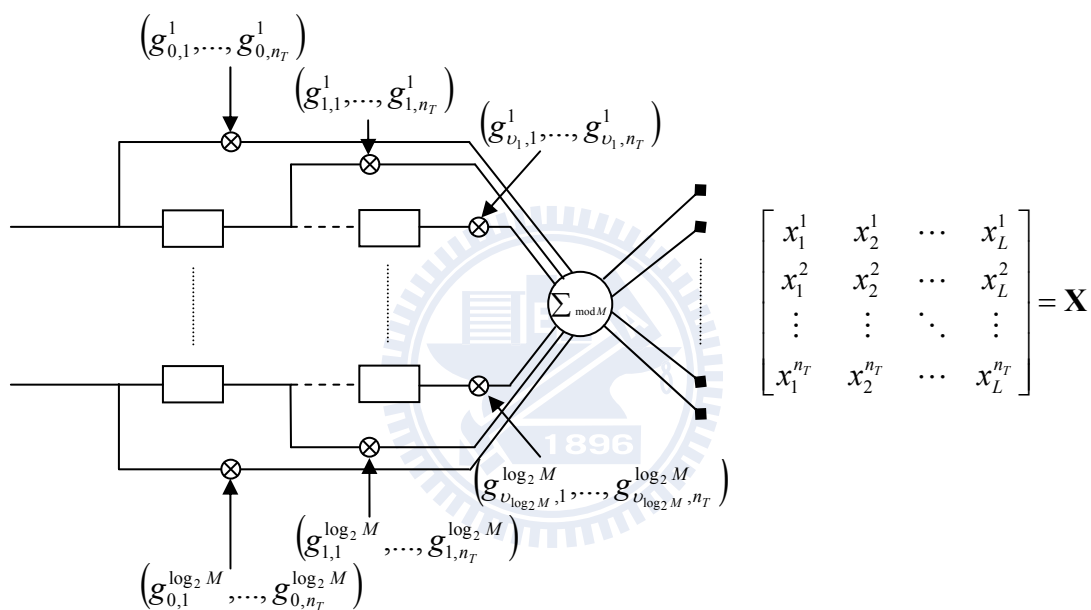


圖 3-1 籬柵空時碼架構圖

(圖 3-1)為籬柵空時碼架構圖，在時間點 t 時，第 k 個輸入流(input stream)的輸入資訊稱為 c_t^k ； $g_{j,i}^k$ 代表第 k 個輸入流的訊號，經過 j 時間後，在第 i 根天線上的衍生係數。籬柵空時碼的天線輸出訊號，可以由下列式子表示：

$$x_t^i = \sum_{k=1}^{\log_2 M} \sum_{j=0}^{v_k} g_{j,i}^k c_{t-j}^k \quad \text{mod } M \quad i = 1, 2, \dots, n_T \quad (\text{Eq 3-1})$$

M 代表籬柵空時碼所使用的調變程度大小(level)， v_k 代表第 k 個輸入流上暫存器的個數。由於輸出與輸入資訊，經由暫存器的影響後，可由籬柵的關係圖建造出來，資訊可以分部到空間與時間上，所以稱之為籬柵空時編碼器。

3.1.2 籬柵空時碼架構特性

從結構看來，籬柵空時碼看似由多個迴旋碼(convolution code)組合而成，也不難發現籬柵空時碼的輸出是 symbols，所以此架構有 trellis code modulation 的味道。在籬柵空實碼中， $\{g\}$ 值占了很重要的因素，因此設計好的 $\{g\}$ 值，對於籬柵空時碼的效能，是非常重要的課題。籬柵空時碼的特性，我們可以從迴旋碼的特性來猜測，列於下方：

- 當 frame size 夠長，相同的編碼率下，memory 的個數愈多，籬柵空時碼的效能愈好；因為 memory 個數愈多，碼與碼之間的距離，愈有機會拉大，保護性會愈高。以 1bit/sec/Hz，2-state 與 4-state 的籬柵空時碼來做比較，圖示於下方：

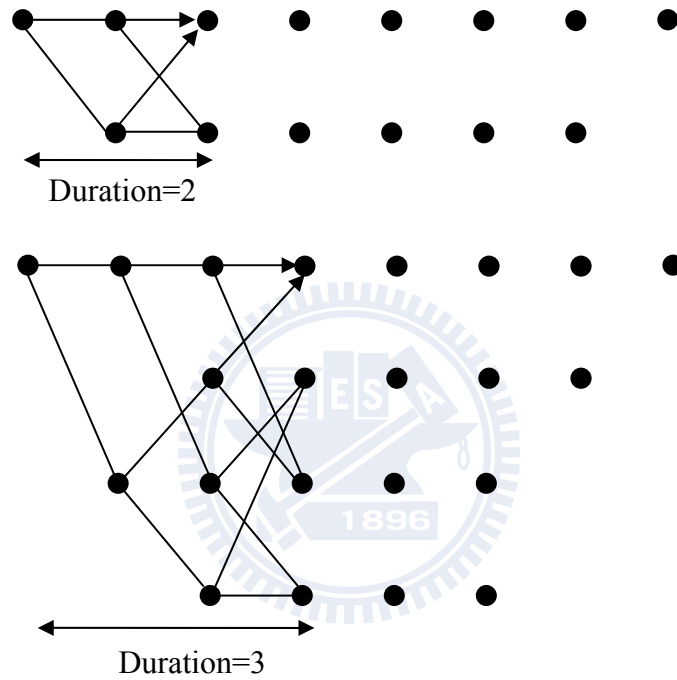


圖 3-2 memory 個數與 coded symbol 間距離的關係

由上方兩圖，我們可以觀察(圖 3-2)上圖在兩個時間點過後，開始有結合的 branch，而(圖 3-2)下圖卻在三個時間點後，才有結合的 branch 發生，所以(圖 3-2 下圖)的編碼中，碼與碼之間的距離，較有機會比(圖 3-2 上圖)的編碼方法還大。但是 memory 的個數愈大，不代表更正能力愈強，memory 個數使用到某種程度時，更正能力進步有限，因為考慮到每個 symbol 所使用的 level 問題，symbol-by-symbol 間的最小距離成長有限。

- 使用籬柵空時碼的 Frame 長度，要注意是否夠長。所謂的夠長，直覺上，表示至少要大於籬柵空時碼的 constraint length；反之，假如小於或等於 constraint length，all zero state terminate 的任務一定無法達成。
- 籬柵空時碼的 diversity，可以由 difference matrix 初略估計。由第二章的 \mathbf{B} 矩陣觀察， \mathbf{B} 矩陣最大的 rank 為 $\min(n_T, L)$ ， L 代表 frame 長度。根據特點(2)， L 的長度又與 memory 的個數有關，所以我們可以知道在籬柵空時碼中，即使有足夠的天線數目，

能支援高 diversity 的傳輸，也需要夠多的 memory 或夠長的 frame 長度來達到所要求的 diversity 表現，下表為一個實際例子，memory 與天線發射分集的關係。

表 3-1 Upper bound of the rank values for STTC

	$n_T=2$	$n_T=3$	$n_T=4$	$n_T=5$	$n_T=6$
$\nu=2$	2	2	2	2	2
$\nu=3$	2	2	2	2	2
$\nu=4$	2	3	3	3	3
$\nu=5$	2	3	3	3	3
$\nu=6$	2	3	4	4	4

由上表顯示，要達到 full diversity 的話，STTC 的 constraint length 至少為 n_T 。

證明：不失其一般性，比較全為零的 information bits 所造出來的 coded symbol 矩陣為 \mathbf{X} 與在 STTC 中其一個 coded symbol 矩陣 $\hat{\mathbf{X}}$ ，它們之間的差，表示如下：

$$\mathbf{X} = \begin{bmatrix} 1 & 1 & \dots & 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 & 1 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 & 1 & \dots & 1 \end{bmatrix} \quad \hat{\mathbf{X}} = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_K^1 & 1 & \dots & 1 \\ x_1^2 & x_2^2 & \dots & x_K^2 & 1 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_1^{n_T} & x_2^{n_T} & \dots & x_K^{n_T} & 1 & \dots & 1 \end{bmatrix}$$

$$\mathbf{X} - \hat{\mathbf{X}} = \begin{bmatrix} 1-x_1^1 & 1-x_2^1 & \dots & 1-x_K^1 & 0 & \dots & 0 \\ 1-x_1^2 & 1-x_2^2 & \dots & 1-x_K^2 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 1-x_1^{n_T} & 1-x_2^{n_T} & \dots & 1-x_K^{n_T} & 0 & \dots & 0 \end{bmatrix}$$

K 代表 constraint length。被比較的 coded symbol 矩陣，是因為我們知道籬柵空時碼的 initial state 為零，至少在 constraint length 之後，會與全為零所製造出來之 coded symbol 的 path 重合。因為上面的差異矩陣要為 full rank，所以 $K \geq n_T$ 。以(圖 3-3)為例，constraint length 為 2，所以此籬柵空時碼編碼器有機會達到 full diversity。

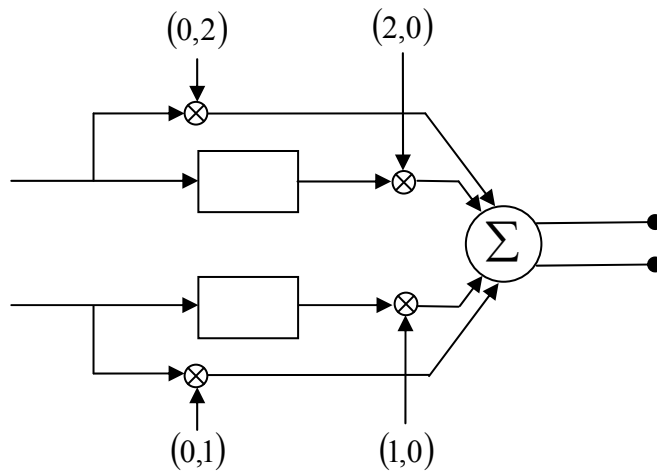


圖 3-3 具延遲效應的籬柵空時碼

- 要達到 full diversity，即是要讓 difference 矩陣達到 full rank，在[2]中，提出一個最簡單的方法— delay diversity，通常達到此目的的編碼，constraint length 為 n_T 、天線數為 n_T ，如(圖 3-3)：天線的係數，隨著 delay，轉移到另一根天線上；如上圖，第二根的天線係數：2、1，經過一個 delay 之後，轉移到第一根天線係數上。這樣的編碼方式，根據上述的 encoder，我們以一個例子說明，假如輸入的資訊為 10,01,11,00,01，輸出的結果，用矩陣表示如下：

$$\begin{bmatrix} x^1 \\ x^2 \end{bmatrix} = \begin{bmatrix} 0,2,1,3,0,\dots \\ 2,1,3,0,1,\dots \end{bmatrix}$$

第二根天線的輸出結果，延遲一個 symbol time 後，洽為第一根天線輸出結果。如此一來，difference 矩陣，必為 full rank。這種具備延遲特性的籬柵空時碼，與多階層空時碼的某一種變形非常相似(Diagonal LST)，不過籬柵空時碼提供了空時碼所強調的 transmit diversity。

- 為了降低解碼複雜度，trellis code 解碼，可以使用 M-algorithm 來減少 decoder 端的解碼複雜度。M-algorithm 的理念就是在每一個 Viterbi 的 time level，只留下 M 個 state，使得下一次延伸出去的 branch 數目減少，如此一來，計算量也會減少。

3.1.3 應用於籬柵解碼的討論

3.1.3.1 Viterbi decoder

對於任意的迴旋碼，Viterbi algorithm 是屬於最佳的 ML 解碼方式。他的精神在於 minimize FER (取最小的 path metric)，而非 minimize BER(minimize BER 的方法有 BCJR...)。Viterbi 的重點如下：

- 必須限定在 all zero state start all、all zero state terminated。
- 解碼複雜度，隨者 constraint length 與 state 個數增加，而成指數成長。
- 在每個 trellis level 把握一個中心原則，說明如下：

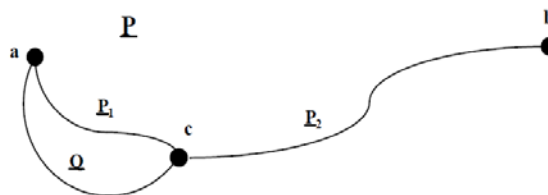


圖 3-4 Viterbi 演算法示意圖

假設 $a \rightarrow b$ 產生最小距離的路徑為 $P(a \rightarrow c \rightarrow b)$ ， c 為路徑中所經過的某個 state，將 P 路徑以 c 為分界點，分為 P_1 和 P_2 ，則 P_1 必屬於 $a \rightarrow c$ 的最短距離。

Proof: (反證法)假設 Q 為 $a \rightarrow c$ 的最短距離， $Q + P_2 \leq P_1 + P_2$ ，最短路徑不為 $P_1 + P_2$ ，與假設矛盾。

3.1.3.2 M-algorithm

簡化 Viterbi 的複雜度，在每一個 trellis level 只留下 M 個 survivor，留下的原則，是根據 path metric 所累積的距離大小做”sorting”，留下最小的 M 個。如此一來，在計算複雜度可以降低；此外，假如 memory 個數愈大的話，可以刪掉的 survivor 愈多，對於高 state 個數的 code，更能有效降低複雜度。注意，當 $M=2^{\text{暫存器個數}}$ ，複雜度跟 optimal Viterbi 相同。

3.1.3.3 Adaptive M-algorithm

雖然 *M-algorithm* 的執行方法既簡單又能大幅降低複雜度，但假如 channel 受到 burst error，在刪除 survivor 時，很容易造成”刪除錯誤”，讓解碼的錯誤率上升。因此我們希望能夠適當的找一個方法，來彌補”簡化複雜度造成錯誤率上升”的犧牲。方法如下，當使用 *M-algorithm* 造成解碼錯誤，在下一階段時，我們採用 *2M-algorithm*，再解碼一次，假如解碼成功，就停止解碼；假如解碼失敗，就再使用 *4M-algorithm* 解碼；直到 *M_{Max}-algorithm* 為止，假如都無法解碼成功的話，才宣告”真正”的解碼失敗。

3.1.3.4 BCJR algorithm

BCJR 的實作方法，先求做 γ 、 β 、 α ，再利用 APP(a posteriori probability) method 去解出 information bits/symbols。

表 3-2 BCJR 參數的計算方法

α	$\alpha_0^*(s) = 0 \quad \text{if } (s = 0),$ $\alpha_0^*(s) = -\infty \quad \text{if } (s \neq 0)$ $\alpha_{t+1}^*(s) = \text{Max}^*(r_t^*(s', s) + \alpha_t^*(s)), \quad t = 0, 1, \dots, N-1$
β	$\beta_{N+1}^*(s) = 0 \quad \text{if } (s = 0),$ $\beta_{N+1}^*(s) = -\infty \quad \text{if } (s \neq 0)$ $\beta_t^*(s) = \text{Max}^*(r_t^*(s', s) + \beta_t^*(s)), \quad t = N-1, N-2, \dots, 0$
γ	$\gamma_t^*(s', s) = p_t(i) \cdot \exp\left(\frac{-d^2(\mathbf{r}_t, \mathbf{x}_t)}{2\sigma^2}\right) \quad i = \text{symbol samples}$
Comments	<p>S 屬於 {下一個時間的 state space}</p> <p>S' 屬於 {這一個時間的 state space}</p> <p>N = coded symbol length</p> $\text{Max}^*(x, y) \equiv \ln(e^x + e^y) = \max(x, y) + \ln(1 + e^{- x-y })$

BCJR 決定每一個 coded symbol output 時，是考慮每一條 transition branch；不是像 Viterbi 一樣，決定 coded symbol output 時，只考慮 survivor path。所以理論上，BCJR 的錯誤率

表現，會比 Viterbi 好。BCJR 計算 max 時，假如沒有加上 log 修正項，會變成 Max-log MAP，Max-log MAP 與 log MAP 可能會有不同的結果。一般來說使用 Max-log MAP 會等效於 Viterbi 解碼。

3.1.4 籬柵空時碼模擬

3.1.4.1 籬柵空時碼不同解碼方式比較

籬柵空時碼中，關於不同解碼方法的表現，以實際模擬驗證。軟體採用 Visual C++ 2005，實驗次數為 $1e7$ 次，每次實驗傳送一個封包，封包內含 260bits，當一 frame 內的一個 bit 錯誤，即算此封包傳送失敗。

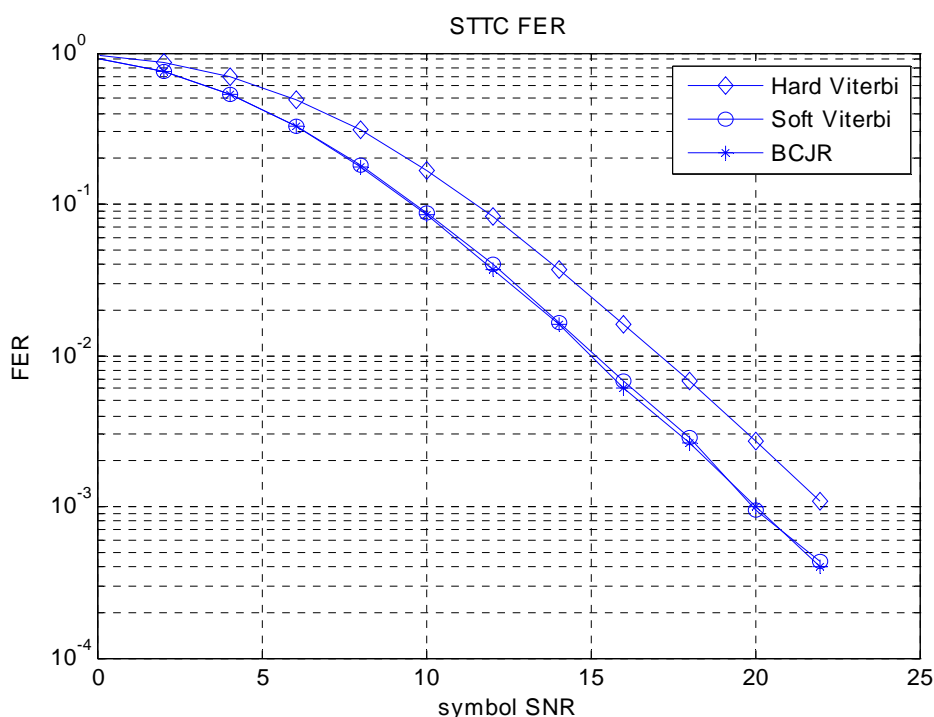


圖 3-5 hard、soft viterbi 與 BCJR decoding 解碼表現比較圖

STTC states	Generator sequence (Source)	rank	determinant	trace
4-states	$g^1=[(02),(20)]$ $g^2=[(01),(10)]$	2	4	4

由上述模擬，我們得知 Soft Decision Viterbi 與 BCJR 的解碼效果，比 Hard decision Viterbi 好。Soft Decision Viterbi 與 BCJR 結果相近的原因，是因為在 MIMO 中，BCJR 的 correct term 很小，導致 BCJR 與 Soft Decision Viterbi 模擬結果相近。不過 BCJR 的特點是考慮事前機率，故很適合需要遞迴解碼的編碼或者不對稱通訊(UEP)使用，例如渦輪空時碼。

3.1.4.2 記憶性與籬柵空時碼的關係

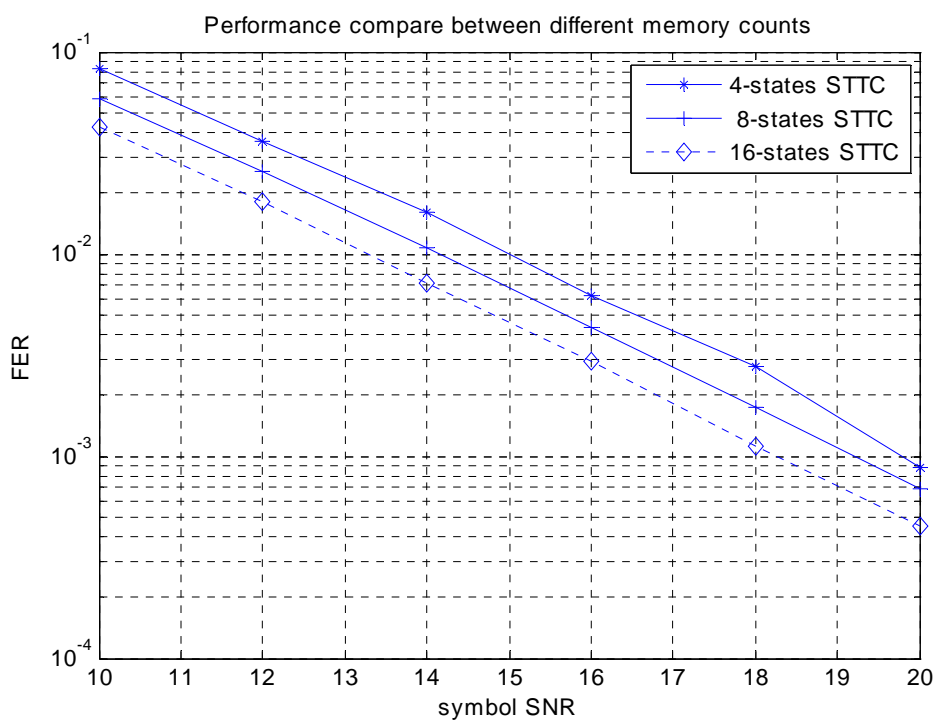


圖 3-6 memory 多寡與錯誤率表現的關係圖

STTC states	Generator sequence (Source)	rank	determinant	trace
4-states	$g^1=[(02),(20)]$ $g^2=[(01),(10)]$	2	4	4
8-states	$g^1=[(02),(20)]$ $g^2=[(01),(10),(22)]$	2	12	8
16-states	$g^1=[(02),(20),(02)]$ $g^2=[(01),(12),(20)]$	2	12	8

增加 centralized STTC encoder 的 memory 個數，等同於增加 state 個數，對於錯誤率的 coding gain 有所增益。

3.1.4.3 使用 M-Algorithm 於維特比解碼

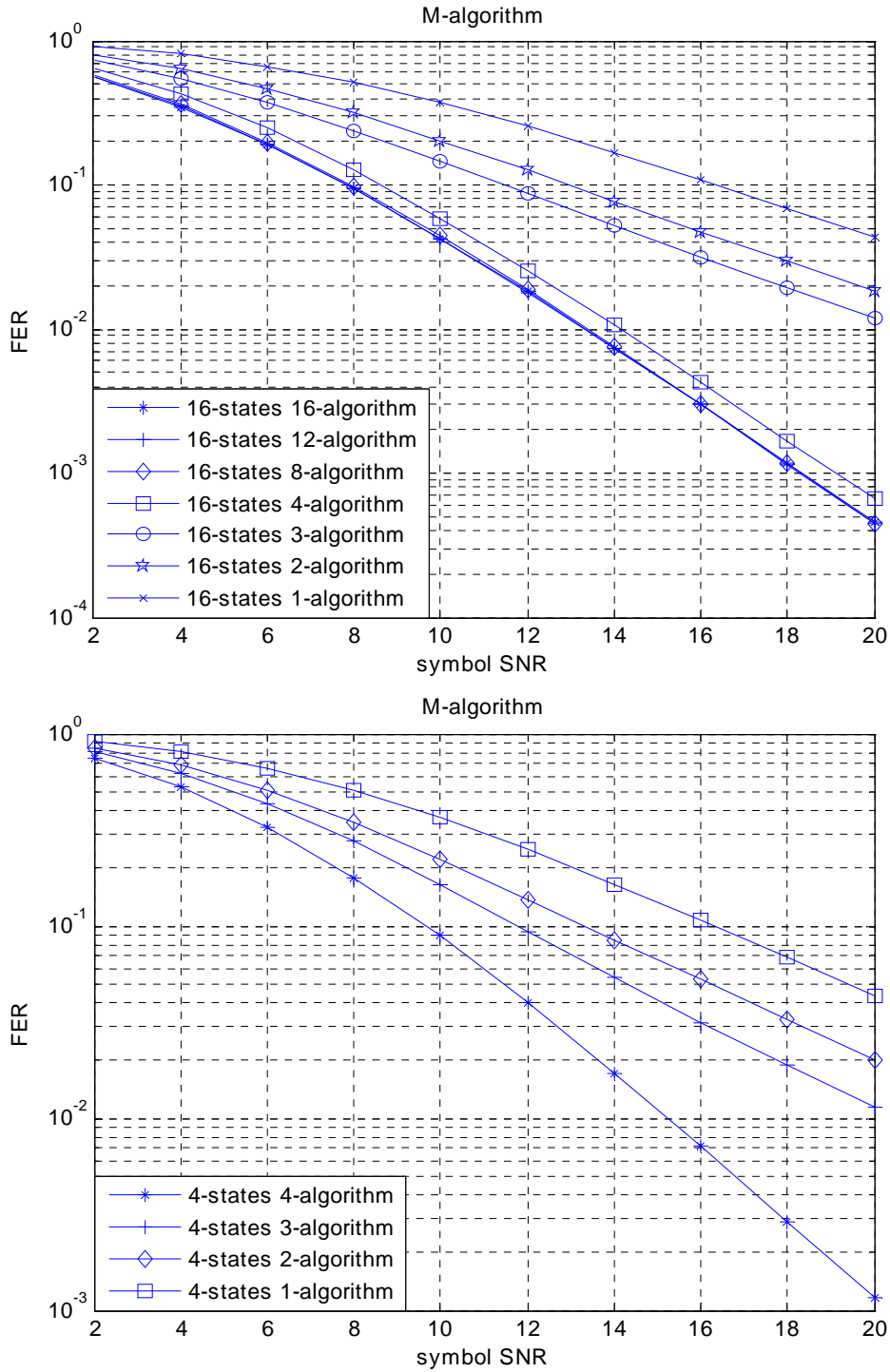


圖 3-7 memory 多寡與錯誤率表現的關係圖

由模擬結果發現，當 M 值愈小時，效能會愈差；但對於高 constraint length 的 code 來說，能夠犧牲的 survivor 愈多。舉例：16-states 的 code，用 8-algorithm 解碼時，效能跟 optimal Viterbi(M=16)差不多；但 4-states 的 code，用 2-algorithm 解碼時，效能就會降低很多。

3.2 區塊空時碼(Space Time Block Code)

區塊空時碼是一種簡單的空時碼實現方法，依靠調變訊號的擺設，即能達到發射分集的目的，在此章節中，我們以最典型的 Alamouti scheme 來介紹。Alamouti scheme 是在兩根傳送天線時，能夠擁有好的頻譜使用效率與簡單的解碼方式。以下是 Alamouti scheme STBC 架構圖：

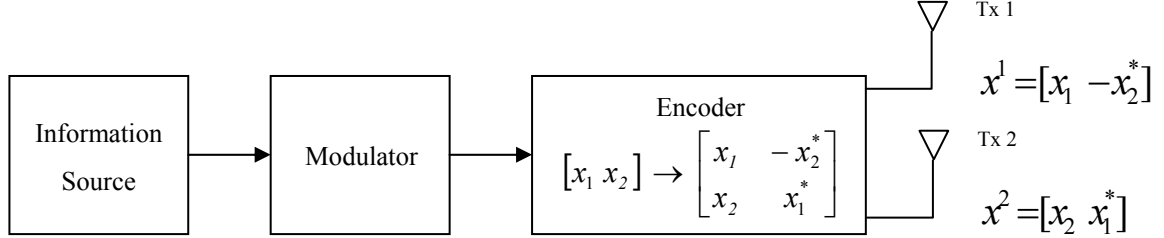


圖 3-8 區塊空時碼架構圖

根據上一章的觀念，我們不難發現，這種傳輸方式，必能保證傳送時，因為它所產生的差異矩陣為 full rank，所以擁有 full transmit diversity。Alamouti 解碼方式使用 ML method，將解碼過程，簡述如下：

假設第 j 根接收天線在時間 t 與 $t+T$ 接收訊號為 r_1^j 、 r_2^j ，

$$r_1^j = h_{j,1}x_1 + h_{j,2}x_2 + n_1^j \quad (\text{Eq. 3-1})$$

$$r_2^j = -h_{j,1}x_2^* + h_{j,2}x_1^* + n_2^j \quad (\text{Eq. 3-2})$$

假設接收端已知通道的值，將訊號作處理，方法如下：

$$\tilde{x}_1 = \sum_{j=1}^{n_R} h_{j,1}^* r_1^j + h_{j,2} (r_2^j)^* = \sum_{i=1}^2 \sum_{j=1}^{n_R} |h_{j,i}|^2 x_i + \sum_{j=1}^{n_R} h_{j,1}^* n_1^j + h_{j,2} (n_2^j)^* \quad (\text{Eq. 3-3})$$

$$\tilde{x}_2 = \sum_{j=1}^{n_R} h_{j,2}^* r_1^j - h_{j,1} (r_2^j)^* = \sum_{i=1}^2 \sum_{j=1}^{n_R} |h_{j,i}|^2 x_i + \sum_{j=1}^{n_R} h_{j,2}^* n_1^j - h_{j,1} (n_2^j)^* \quad (\text{Eq. 3-4})$$

接著，可以使用 hard decision 來決定 \hat{x}_1 、 \hat{x}_2

$$\hat{x}_1 = \underset{\hat{x}_1 \in S}{\operatorname{argmin}} \left[\left(\sum_{j=1}^{n_R} (|h_{j,1}|^2 + |h_{j,2}|^2) - 1 \right) |\hat{x}_1|^2 + d^2(\tilde{x}_1, \hat{x}_1) \right] \quad (\text{Eq. 3-5})$$

$$\hat{x}_2 = \underset{\hat{x}_2 \in S}{\operatorname{argmin}} \left[\left(\sum_{j=1}^{n_R} (|h_{j,1}|^2 + |h_{j,2}|^2) - 1 \right) |\hat{x}_2|^2 + d^2(\tilde{x}_2, \hat{x}_2) \right] \quad (\text{Eq. 3-6})$$

其中， S 代表所有可能的 modulation 訊號集合，假設 modulation 訊號是採用 M-PSK 的

話，那麼上面的方法，可以簡化成下方式子，也就是區塊空時碼解碼簡單的原因。

$$\hat{x}_1 = \arg \min_{\hat{x}_1 \in S} d^2(\tilde{x}_1, \hat{x}_1) \quad (\text{Eq. 3-7})$$

$$\hat{x}_2 = \arg \min_{\hat{x}_2 \in S} d^2(\tilde{x}_2, \hat{x}_2) \quad (\text{Eq. 3-8})$$

在兩根天線的傳輸方法中，假如使用 ML 的解碼方法，它的結果剛好即是 MRC(Maximum Ratio Combining)，以 2X1 MIMO 系統為例子，接收端所做的訊號處理如下：

$$\tilde{x}_1 = h_1^* r_1 + h_2 r_2^* = (|h_1|^2 + |h_2|^2) x_1 + h_1^* n_1 + h_2 n_2^* \quad (\text{Eq. 3-9})$$

$$\tilde{x}_2 = h_2^* r_1 - h_1 r_2^* = (|h_1|^2 + |h_2|^2) x_2 - h_1 n_2^* + h_2^* n_1 \quad (\text{Eq. 3-10})$$

顯而易見，經過訊號處理過後，將通道的相位移除，即是 MRC 的結果。

3.3 多層次空時碼(Layered space time code)

籬柵空時碼(Space time trellis code)複雜度隨著資料傳輸速度成指數成長，因此有學者[1]提出階層式空時碼(layered space time code)，階層式空時碼主要的特點是架構較為簡單，並能夠接近 MIMO 系統的 capacity。下圖為階層式空時碼的架構圖：



圖 3-9 多層次空時碼架構圖

由架構圖可以看出，將輸入資料，先做分流，此部分與 OFDM 有異曲同工之妙；利用分流，加速資料傳送速度；傳統直覺上，階層式空時碼與先前的籬柵空時碼對付的問題不同，階層式空時碼主要強調增加資料速度(multiplexing)，籬柵空時碼則是強調增加對資料的保護性(diversity)，由前篇章節可得知，這兩種特性是相互消長，不過同為空時碼的應用。階層式空時碼的解碼方法，在此簡單介紹兩種經典方法

1) QR Decomposition Interference Suppression Combined with Interference Cancellation

假設接收端已知通道狀況下，將通道做 QR 分解， \mathbf{U}_R 為 unitary 矩陣

$$\mathbf{H} = \mathbf{U}_R \mathbf{R} \quad (\text{Eq. 3-11})$$

$$\mathbf{R} = \begin{bmatrix} (R_{1,1})_t & (R_{1,2})_t & \cdots & (R_{1,n_r})_t \\ 0 & (R_{2,1})_t & \cdots & (R_{2,n_r})_t \\ 0 & 0 & \cdots & (R_{3,n_r})_t \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & (R_{n_r,n_r})_t \end{bmatrix}$$

將接收到的訊號，乘上 \mathbf{U}_R^T ，得到下列結果：

$$\mathbf{U}_R^T \mathbf{r} = \mathbf{U}_R^T \mathbf{H} \mathbf{x} + \mathbf{U}_R^T \mathbf{n} \rightarrow \mathbf{y} = \mathbf{R} \mathbf{x} + \mathbf{n}' \quad (\text{Eq. 3-12})$$

分析 \mathbf{y} 矩陣內的每個元素：

$$y_t^i = (R_{i,i})_t x_t^i + n_t^i + \sum_{j=i+1}^{n_r} (R_{i,j})_t x_t^j \quad (\text{Eq. 3-13})$$

因為 \mathbf{R} 為上三角矩陣，所以我們可以先觀察 $i=n_r$ 的情況：

$$y_t^{n_r} = (R_{n_r,n_r})_t x_t^{n_r} + n_t^{n_r} \rightarrow \hat{x}_t^{n_r} = q \left(\frac{y_t^{n_r}}{(R_{n_r,n_r})_t} \right) \rightarrow \text{解出 } \hat{x}_t^{n_r}, \text{ 帶入 } i=n_r-1 \text{ 的情況}$$

↓

$$y_t^{n_r-1} - (R_{n_r,n_r-1})_t x_t^{n_r} - (R_{n_r,n_r})_t x_t^{n_r} = (R_{n_r-1,n_r-1})_t x_t^{n_r-1} + n_t^{n_r-1} \rightarrow$$

$$\hat{x}_t^{n_r-1} = q \left(\frac{y_t^{n_r-1} - (R_{n_r,n_r-1})_t x_t^{n_r} - (R_{n_r,n_r})_t x_t^{n_r}}{(R_{n_r-1,n_r-1})_t} \right) \rightarrow \text{解出 } \hat{x}_t^{n_r-1}, \text{ 帶入 } i=n_r-2 \text{ 的情況}$$

↓

$$y_t^i - \sum_{j=i+1}^{n_r} (R_{i,j})_t \hat{x}_t^j = (R_{i,i})_t x_t^i + n_t^i \rightarrow \hat{x}_t^i = q \left(\frac{y_t^i - \sum_{j=i+1}^{n_r} (R_{i,j})_t \hat{x}_t^j}{(R_{i,i})_t} \right) \rightarrow \text{解出 } \hat{x}_t^i, \text{ 帶入 } i=i-1 \text{ 的情況}$$

↓

$$\hat{x}_t^1 = q \left(\frac{y_t^1 - \sum_{j=2}^{n_r} (R_{1,j})_t \hat{x}_t^j}{(R_{1,1})_t} \right) \rightarrow \text{解出 } \hat{x}_t^1$$

用此方法解碼時，最大的壞處是會有 error propagation；假如第一個 symbol ($x_t^{n_r}$) 沒有解碼正確，很可能造成後面 symbol 解碼的正確率大幅下降；針對此問題，解決的方式，可以根據通道的增益，先解通道好的 symbol，更深入的問題，請參考相關的代表性論文 [9]。

2) Interference MMSE Suppression Combined with Interference Cancellation

將 MMSE 套入 MIMO 系統中，計算以下式子

$$\min E \left\{ (\mathbf{x} - \mathbf{w}^H \mathbf{r})^2 \right\} \quad \text{where } \mathbf{w}^H = [\mathbf{H}^H \mathbf{H} + \sigma^2 \mathbf{I}_{n_r}]^{-1} \mathbf{H}^H \quad (\text{Eq. 3-14})$$

令 \mathbf{w}_i^T 代表 \mathbf{w}^T 第 i 列的向量，因此 \hat{x}_i^t 的估測方法： $\hat{x}_i^t = q(\mathbf{w}_i^H \mathbf{r})$

上述的方法，物理意義上，猶如通訊系統的 equalizer，將訊號間的 interference 壓抑掉，接著我們結合 interference cancellation。

令通道矩陣 \mathbf{H} 的第 i 行的向量為 \mathbf{h}_i ，假使我們先估計出 \hat{x}_i^t 時，我們可從接收訊號扣除 $\hat{x}_i^t \mathbf{h}_i$ ，再解其他的訊號，這個動作即稱為 interference cancellation。一般來說，解碼的順序為 \hat{x}_n^t 、 \hat{x}_{n-1}^t 、...、 \hat{x}_1^t ，隨著通道狀況的好壞，不一定要照這樣的順序。最後要注意的是，將 \hat{x}_i^t 解出來，再解下一個 \hat{x}_{i-1}^t 時，必須將 \mathbf{h}_i 從 \mathbf{H} 捨掉，因此定義下列矩陣：

$$\mathbf{H}_d^{i-1} = \begin{bmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,i-1} \\ h_{2,1} & h_{2,2} & \cdots & h_{2,i-1} \\ \vdots & \vdots & \vdots & \vdots \\ h_{n_r,1} & h_{n_r,2} & \cdots & h_{n_r,i-1} \end{bmatrix}$$

Linear MMSE Suppression and Successive Cancellation 演算法：

Set $i = n_r$, $r^{n_r} = r$

while $i \geq 1$

{

$$\mathbf{w}^H = [\mathbf{H}^H \mathbf{H} + \sigma^2 \mathbf{I}_{n_r}]^{-1} \mathbf{H}^H$$

$$y_i^t = \mathbf{w}_i^H r^i$$

$$\hat{x}_i^t = q(y_i^t)$$

$$\mathbf{r}^{i-1} = \mathbf{r}^i - \hat{x}_i^t \mathbf{h}_i$$

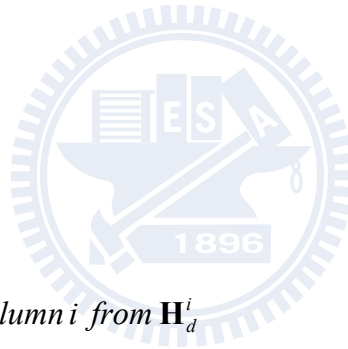
compute \mathbf{H}_d^{i-1} by deleting column i from \mathbf{H}_d^i

$$\mathbf{H} = \mathbf{H}_d^{i-1}$$

$$i = i - 1$$

}

由上面的敘述可知，想要在解碼端作簡化，我們可以將複雜的解碼方式，分解成多個容易的解碼方式。



第四章 空時籬柵碼之搜尋演算法

(Space Time Trellis Code Search (Pragmatic STTC + Viterbi Search))

在前篇的介紹中，我們得知設計一個好的空時碼，大致上分為下列幾個準則：

- 1) Rank criterion
- 2) Determinant criterion
- 3) Trace criterion

MIMO 系統中，當天線數夠多時，設計籬柵空時碼，使用 trace 準則，較為適當；更進一步解釋，就是看 coded symbols 間的距離，來決定 code 的好壞。對於籬柵空時碼來說，它的架構實為複雜，尋找 code 的方法，已有多種系統化的方式提出，但大多數仍存在許多問題，比如說全域搜尋法花費很多時間、沒有完全運用到 trellis 的特性。對於籬柵空時碼這種繁複的編碼架構與搜尋過程，我們提出簡化的方法，讓搜尋的時間大幅降低，以便往後研究者縮短模擬時間。以下由簡入深介紹搜尋的方式。

4.1 全域搜尋演算法(exhaustive search)

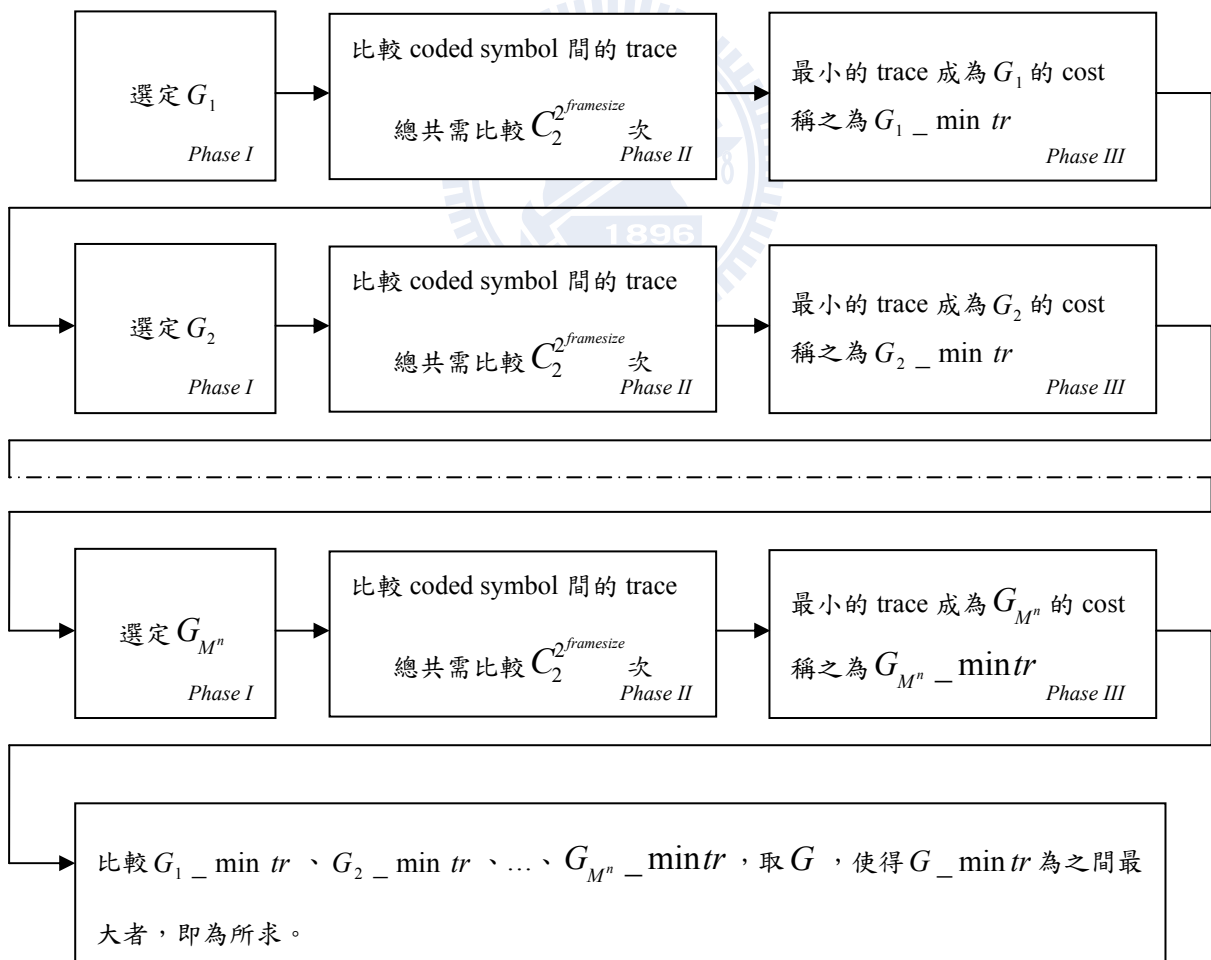


圖 4-1 全域搜尋演算法流程圖

尋找籬柵空時碼時，把所有可能的衍生係數 (generate sequence)都試一遍，每次選定一組衍生係數時，即是選定一組籬柵空時碼編碼器的編碼方式(Phase I)，接下來把所有可能的 information bits 輸入已選定的籬柵空時碼編碼器中，產生 codes symbols，計算 coded symbols 間彼此的 trace(Phase II)；coded symbols 間，最小的 trace，代表此衍生係數的 cost(Phase III)。假設衍生係數有 n 個，M-PSK 籬柵空時碼狀況下，要計算 M^n 組衍生係數；另外，每次求得最小的 trace 前，需要計算 $C_2^{2^{framesize}}$ 組 coded symbol 的 trace。以上將搜尋的過程用圖示表達出來。

4.2 基於天線個數之遞迴演算法(Pragmatic STTC)

Proposition：對於 n_T-1 根天線，採用 trace criterion 所設計出來的 optimal code，其係數對應如下：

$$g_{j,p}^k \quad k=1,2,\dots,m \quad j=1,2,\dots,v_k \quad i=1,2,\dots,n_T-1$$

則必可對 n_T 根天線的 MIMO 系統，同樣造出 optimal code，其係數如下：

$$\tilde{g}_{j,p}^k \quad k=1,2,\dots,m \quad j=1,2,\dots,v_k \quad i=1,2,\dots,n_T \quad \text{where } \tilde{g}_i^k = \tilde{g}_{j,i}^k \text{ for } i=1,2,\dots,n_T-1$$

Proof.

step 1.

假設對於 n_T-1 根天線來說，我們已經用有 optimal 的 encode 方法

$$x_i^i = \sum_{k=1}^m \sum_{j=0}^{v_k} g_{j,i}^k c_{t-j}^k \quad \text{mod } M \quad i=1,2,\dots,n_T-1 \quad (\text{Eq. 4-1})$$

$$\hat{x}_i^i = \sum_{k=1}^m \sum_{j=0}^{v_k} \hat{g}_{j,i}^k \hat{c}_{t-j}^k \quad \text{mod } M \quad i=1,2,\dots,n_T-1 \quad (\text{Eq. 4-2})$$

因為依照 trace criterion 設計，故以下式子成立：

$$\min \left\{ \sum_{t=1}^L \sum_{i=1}^{n_T-1} |x_t^i - \hat{x}_t^i|^2 \mid C \neq \hat{C} \right\} \geq \min \left\{ \sum_{t=1}^L \sum_{i=1}^{n_T-1} |y_t^i - \hat{y}_t^i|^2 \mid C \neq \hat{C} \right\} \quad (\text{Eq. 4-3})$$

y_t^i, \hat{y}_t^i 是任意籬柵空時碼係數所製造出來的 encode symbol。

step 2.

對於 n_T 根天線來說， n_T 與 n_T-1 根天線的差別，在於多了 $x_t^{n_T}$ 項：

$$x_t^{n_T} = \sum_{k=1}^m \sum_{j=0}^{v_k} g_{j,n_T}^k c_{t-j}^k \quad \text{mod } M$$

$$\hat{x}_t^{n_T} = \sum_{k=1}^m \sum_{j=0}^{v_k} g_{j,n_T}^k \hat{c}_{t,j}^k \pmod M$$

$$\min \left\{ \sum_{t=1}^L \sum_{i=1}^{n_T-1} |x_t^i - \hat{x}_t^i|^2 \middle| C \neq \hat{C} \right\} + \sum_{t=1}^L |x_t^{n_T} - \hat{x}_t^{n_T}|^2 \geq \min \left\{ \sum_{t=1}^L \sum_{i=1}^{n_T-1} |y_t^i - \hat{y}_t^i|^2 \middle| C \neq \hat{C} \right\} + \sum_{t=1}^L |x_t^{n_T} - \hat{x}_t^{n_T}|^2 \quad \text{式子}$$

仍然成立 $\therefore \min \left\{ \sum_{t=1}^L \sum_{i=1}^{n_T} |x_t^i - \hat{x}_t^i|^2 \middle| C \neq \hat{C} \right\} \geq \min \left\{ \sum_{t=1}^L \sum_{i=1}^{n_T} |y_t^i - \hat{y}_t^i|^2 \middle| C \neq \hat{C} \right\}$ 所以設計 n_T 根天線

optimal code 時，可以沿用設計 n_T-1 根天線中，所用的係數。

step 3.

反之，當使用 trace criterion 時，設計 n_T 根天線的 optimal code 時；假使某個時間點，只需要用到 n_T-1 根天線傳送時，我們可以直接刪掉關於第 n_T 根天線的所有係數(註)。

這個部分，簡化全域搜尋法中 phase 1 的演算法複雜度，使得對於搜尋 G 值時，不需因為天線過多，而增加搜尋時間；搜尋 n_T 根天線的籬柵空時碼最佳解，只要基於 n_T-1 根天線的最佳解即可，如此一來，隨著傳送天線愈多，對於擁有相同籬柵結構的籬柵空時碼，搜尋的時間仍然相同。

註：一定要刪掉 n_T 根天線的相關係數嗎？假使在設計 code 的流程是按照 step1、step2 步驟走的話，從第 1、2、...、 n_T 根依序設計的話，那麼刪掉第 n_T 根天線的所有相關係數是合理的。假使不是依序設計的話，刪除係數時，就不一定是刪掉第 n_T 根天線的係數。

4.3 應用於空時籬柵碼搜尋之維特比演算法

(STTC code search using Viterbi algorithm)

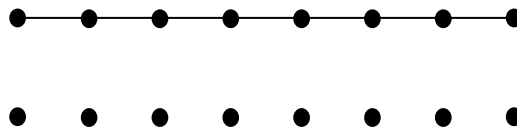
當我們選定一組籬柵空時碼編碼器時，我們要針對所有可能 information bits 產生出來的 coded symbols，兩兩做距離比較；此部分，目的是為了簡化比較時的複雜度。Viterbi algorithm 是降低 ML algorithm 的一種方法，我們將運用 trellis 的特性，用 Viterbi algorithm 爭取時間，來快速獲得結果。方法如下：

Step 1:

選定一組籬柵空時碼時，我們隨即可以製造所對應的 trellis。

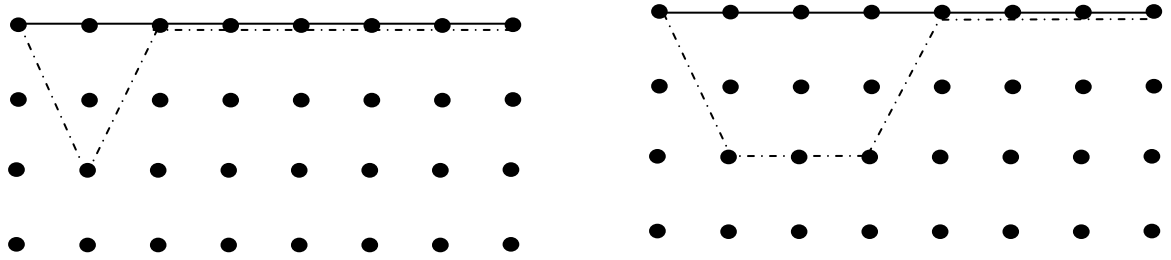
Step 2:

於時間 t ，從 State 0 出發，輸入一組 information bits \mathbf{X} 至籬柵空時碼編碼器，在籬柵上，對應一條 path，獲得一組 coded symbol 的組合。例如：information bits=(0000000)，從 state0 出發，得到的 path 如下(實線表示)



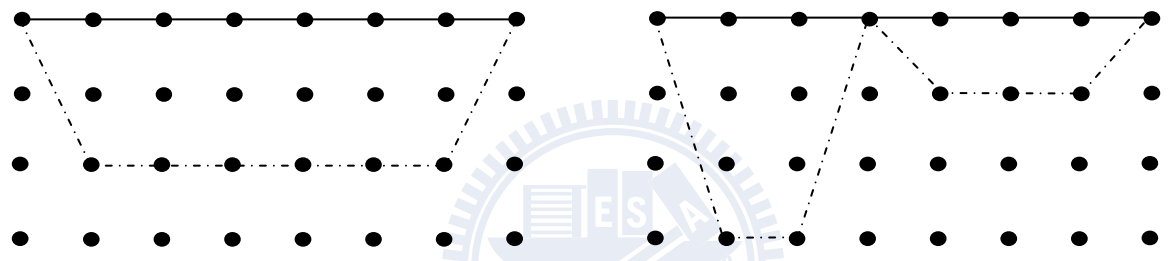
不失其一般性，產生 error 的 codes 為 $\{\hat{\mathbf{X}}\}$ ，在某一時間 $\{\hat{\mathbf{X}}\}$ 於 State 0 開始與 \mathbf{X} 分離，注意這裡所定義的 $\{\hat{\mathbf{X}}\}$ 為一個集合，代表所有的 error codes 都是在 State 0 開始與 \mathbf{X} 有分歧。

如下圖所示： $\{\hat{\mathbf{X}}\}$ 用虛線表示， \mathbf{X} 用實線表示。



圖(一)

圖(二)



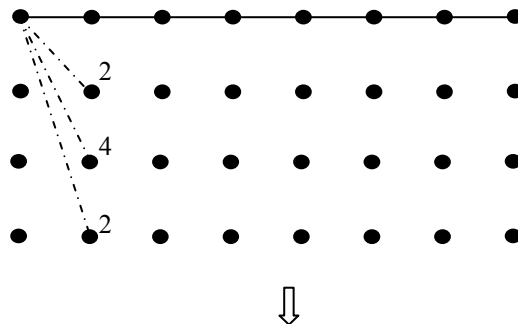
圖(三)

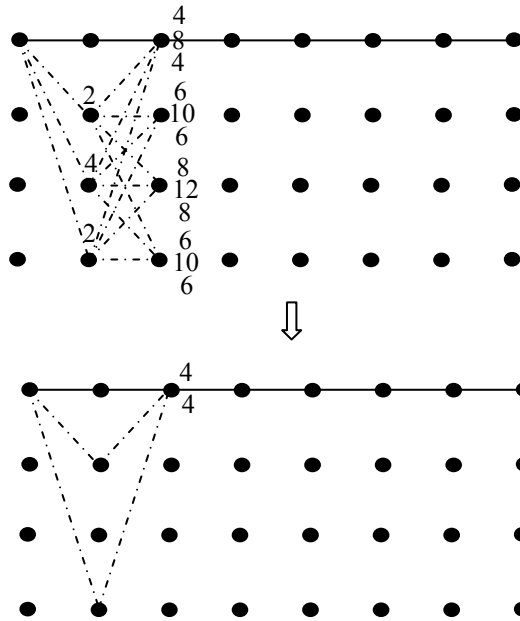
圖(四)

Step 3:

經過 constraint length 時間過後， $\{\hat{\mathbf{X}}\}$ 裡存在與 \mathbf{X} 相交會，並存在與 \mathbf{X} 的歐式距離很小的 error codes，如上圖一。此外，在比較的過程中，我們盡可能避免不必要的”兩兩比較”。

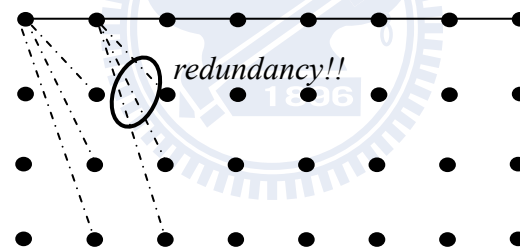
使用 VA，即可達到我們的需求。在此計算距離時，皆是計算 $\{\hat{\mathbf{X}}\}$ 與 \mathbf{X} 的距離。如下圖所示：





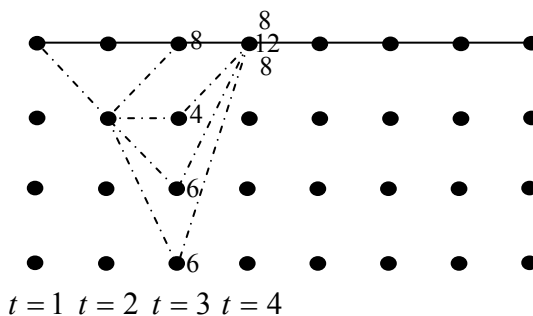
經由上面幾個簡單的步驟，很快就可以找到 STTC 的最小 trace，不必將全部的 coded symbol 做兩兩比較。在執行 Viterbi 的過程中，我們必須注意下列準則：

- 在 t 時間從 state 0 發散，在 $t+1$ 、 $t+2$... 時，不需再計算從 state 0 發散出去的 branch，因為這個動作與在時間 t 從 state 0 發散的作用相同，將會得到相同的結果。如下圖，列舉一個不必要的例子：



- 我們期望發散出去的 path $\{\hat{\mathbf{X}}\}$ ，能夠盡快與 \mathbf{X} 再結合，這樣才能盡快計算出最短距離。

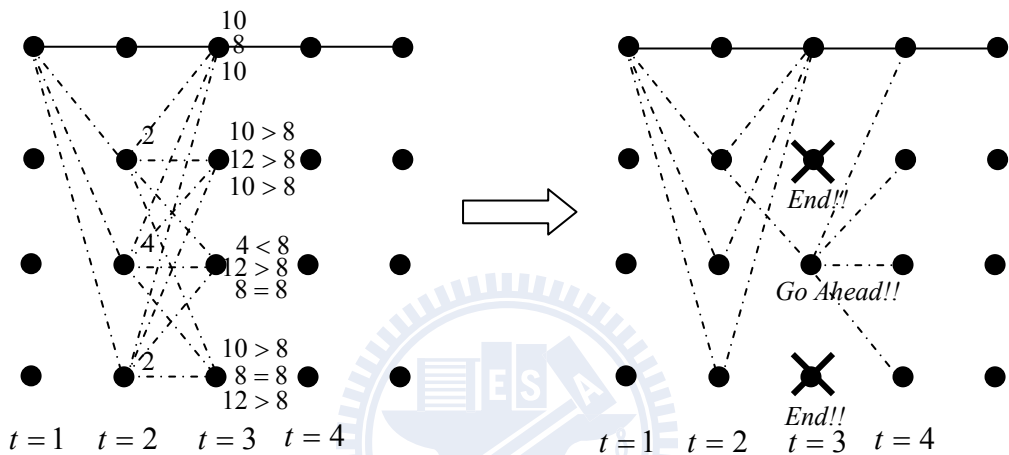
所以當 $\{\hat{\mathbf{X}}\}$ 與 \mathbf{X} 結合在一起的時候，就可以決定可能的最小距離。尚未結合在一起的 path，儘管此 path 比目前暫存的最小距離短，但不能用它來決定最小距離。因為後續延伸的 branch 仍有可能影響結果。以下面的例子，來觀察：



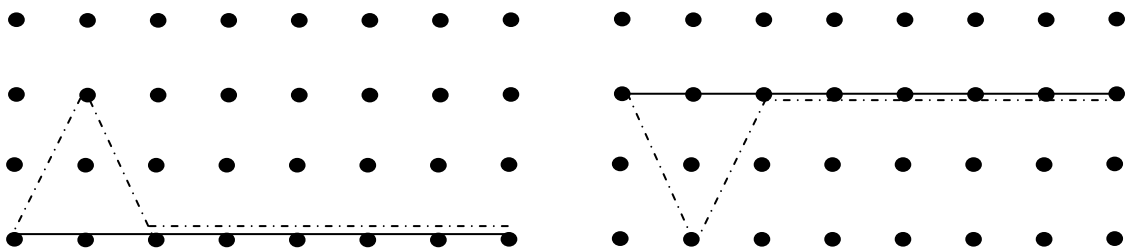
在 $t=3$ 時，與 \mathbf{X} 結合的 branch 中，最小的為 8；state 1、state 2、state 3 的 survivor

與 \mathbf{X} 距離分別為 4、6、6，以上三個值，皆小於目前最小的 trace $\rightarrow 8$ ，但我們不可將這些值(例如：4)，誤以為是最小的 trace。一切的值要在與 \mathbf{X} 結合後，才能做決定；例如本來在 $t=3$ 的所有 survivor path，在 $t=4$ 與 \mathbf{X} 結合後，皆大於或等於 8，假如在 $t=3$ 即決定最小 trace 為 4，就會出錯。一定要與 \mathbf{X} 結合之後，才能決定最小 trace 的原因，是由於我們預先知道 STTC 的編碼方式，在籬柵中最後勢必走到 all zero state，碼與碼間，在籬柵上所代表的路徑一定會有交會重疊的時候，假如還沒有與 \mathbf{X} 結合之前，就判斷碼與碼間的 trace，是不合理的做法，甚至不存在此錯誤的 trace 值。

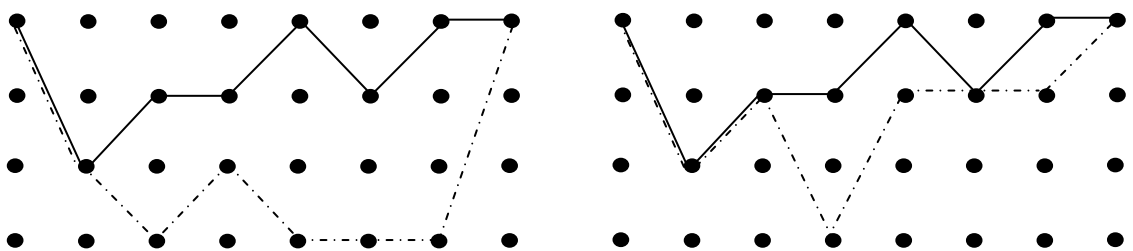
- 當 survivor 的 cost 大於目前所決定出的最小 trace，我們就不必對此 survivor 做延伸了。如下圖所示



不過上面，僅單究對 State 0 發散的 $\{\hat{\mathbf{X}}\}$ 做處理，還要考慮其他狀況，比如從 State 1、State 2、State 3 發散的 $\{\hat{\mathbf{X}}\}$ 做處理。如下面的例子：



所以我們必須回到 step 2，對於 \mathbf{X} 與 $\{\hat{\mathbf{X}}\}$ 分別在 State 1、State 2、State 3 發散的情況，做討論；此外，對於非 all zero input 的 information bits 也要做比較，如下圖。



利用籬柵的特性，使用 Viterbi 演算法去搜尋最佳籬柵空時碼編碼方式，簡化全域搜尋方法中，phase 2 的複雜度，大幅加速搜尋的速度。

4.4 實際模擬驗證

以下利用 Visual c++ 2005，模擬搜尋空時籬柵碼的演算法，比較其效能。搜尋目標為：QPSK、3 根傳送天線、4 個 states 的空時籬柵碼，根據 trace 準則，結果如下：

表 4-1 使用 Pragmatic+Viterbi 演算法於搜尋碼上之驗證結果

搜尋演算法	No.1 實驗	No.2 實驗	No.3 實驗	平均時間
全域搜尋	60768922 msec	60843563 msec	60845922 msec	60819469 msec
Only Viterbi	8891641 msec	8888859 msec	8885062 msec	8888521 msec
Only Pragmatic	953 msec	938 msec	953 msec	948 msec
Pragmatic +Viterbi	156 msec	172 msec	156 msec	161 msec

所搜尋空時籬柵編碼方式，

由模擬結果驗證，當上述介紹的所有方法用於碼搜尋，與全域搜尋法比較時，效率會增加 1e5 倍！

4.5 應用於空時籬柵碼 Determinant Criterion 搜尋之維特比演算法問題討論

參考(Eq 2-20)，矩陣運算具有下列特性：

$$\mathbf{B}=[\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_{\text{framesize}}] \quad \mathbf{b}_t = [x_t^1 - \hat{x}_t^1, x_t^2 - \hat{x}_t^2, \dots, x_t^{n_r} - \hat{x}_t^{n_r}]^T$$

$$\mathbf{A}(\mathbf{X}, \hat{\mathbf{X}}) = \mathbf{B} \cdot \mathbf{B}^H = \sum_{t=1}^{\text{framesize}} \mathbf{b}_t \cdot \mathbf{b}_t^H \quad \rightarrow \quad \det(\mathbf{A}) = \det(\mathbf{B} \cdot \mathbf{B}^H) = \det\left(\sum_{t=1}^{\text{framesize}} \mathbf{b}_t \cdot \mathbf{b}_t^H\right)$$

又得知 $\mathbf{b}_t \cdot \mathbf{b}_t^H$ 為 Hermitian 和 Nonnegative 矩陣，根據[15]得知，假如 $\mathbf{b}_t \cdot \mathbf{b}_t^H$ 為 Hermitian 和 Nonnegative 矩陣，則下列不等式成立：

$$\det\left(\sum_{t=1}^K \mathbf{b}_t \cdot \mathbf{b}_t^H\right) \leq \det\left(\sum_{t=1}^{K+1} \mathbf{b}_t \cdot \mathbf{b}_t^H\right) \quad (\text{Eq 4-4})$$

因前篇章節將 trace criterion 用於 Viterbi 演算法中，所以吾人也試著想將 determinant criterion 用於 Viterbi 演算法中，但發現有些問題；在此節中，先介紹吾人使用的演算法，再去探究其演算法問題所在。

4.5.1 應用於空時籬柵碼 Determinant Criterion 搜尋之維特比演算法

在傳統 Viterbi 演算法中，計算的 cost 為距離；本節中，取而代之的 cost 是 determinant。由於空時碼矩陣中，具備(Eq 4-4)的特性，所以初步認為， \mathbf{B} 矩陣的 determinant 值，會隨著時間增長而變得愈來愈大。為了快速搜尋到最小的 determinant 值，根據前章節敘述，Viterbi 是一種與 ML 效能相當，但效率更快的演算法。當 Viterbi 執行 Add-Compare-Select

時，作法如下：

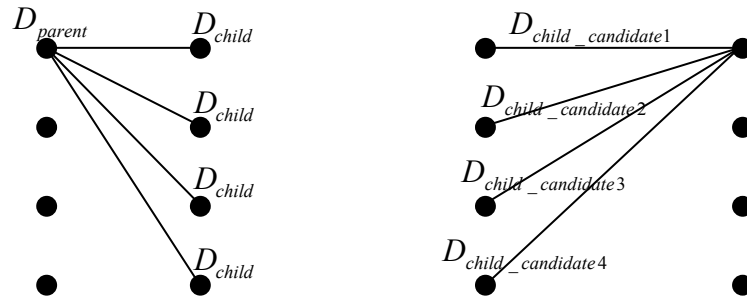


圖 4-2 determinant criterion 應用於 Viterbi 演算法

- Add: Viterbi 執行 add 時， $\mathbf{D}_{parent} = \sum_{t=1}^K \mathbf{b}_t \cdot \mathbf{b}_t^H$ ，又用 $\mathbf{b}_t \cdot \mathbf{b}_t^H$ 代表每一條 branch，所以 $\mathbf{D}_{child} = \mathbf{D}_{parent} + \mathbf{b}_{t+1} \cdot \mathbf{b}_{t+1}^H$ ，參考(圖 4-2 左)。
- Compare and Select: Viterbi 在執行 Compare and Select 的時間點，為存在一條以上的 branch 合併於相同的 state 時。當多條 branch 合併於同一個 state 時，Viterbi 演算法需選擇一條 cost 最小的 branch 保留下來。以(圖 4-2 右)為例，有四條 branch 合併於同一個 state，所以我們選取最小的 $\det(\mathbf{D}_{child_candidate})$ 。

4.5.2 應用於空時籬柵碼 Determinant Criterion 搜尋之維特比演算法問題

- 參考(圖 4-2 右)，當 $\mathbf{D}_{child_candidate1}$ 與 $\mathbf{D}_{child_candidate2}$ 的 determinant 相同時。Viterbi 在選擇 branch 間最小的 cost 時，選 $\mathbf{D}_{child_candidate1}$ 或 $\mathbf{D}_{child_candidate2}$ 皆符合 Viterbi 演算法的原則。假使之後產生較小 determinant 的 code 是由 $\mathbf{D}_{child_candidate1}$ 計算出來(Eq 4-4)，但我們卻在 Viterbi 演算法留下 $\mathbf{D}_{child_candidate2}$ ，就有可能在之後的計算中，產生錯誤。
- 為解決此問題，已有學者提出 Viterbi 以外的解決方法，可以參考[15]，用樹狀圖搜尋。

第五章 合作式通訊的應用

通訊系統上，終端設備的花費愈低，愈能吸引大眾，尤其是手持設備上因尺寸與成本的問題，顯得更為明顯。合作式通訊的提出，解決終端設備只有單一天線的問題，它能藉由鄰近的天線，一起合作，幫忙做訊號傳輸，達到 MIMO 的效果，並節省用戶端電源使用的花費。根據[3]，protocol 大致上分為 Amplify and Forward、Decode and Forward，其他種的 protocol 也依循這兩種方式做延伸。Amplify and Forward 機制中，Relay 端僅對接收訊號作正規化；Decode and forward 機制中，Relay 端對接收訊號做解碼之後，再編碼送出。

此外，合作式通訊中，接取技術有 TDMA、FDMA、SDMA，近期空時編碼技術廣為流行，在本論文的合作式系統架構中，接取技術結合 SDMA、TDMA，將 space time trellis code、Alamouti code 應用於其中，觀察這些方法在合作式通訊上的結果。Space Time Code 能夠解決傳統 repetition based 合作式通訊，傳輸效率不好的問題。

5.1 Cooperative communication protocol 簡介

5.1.1 Amplify and Forward

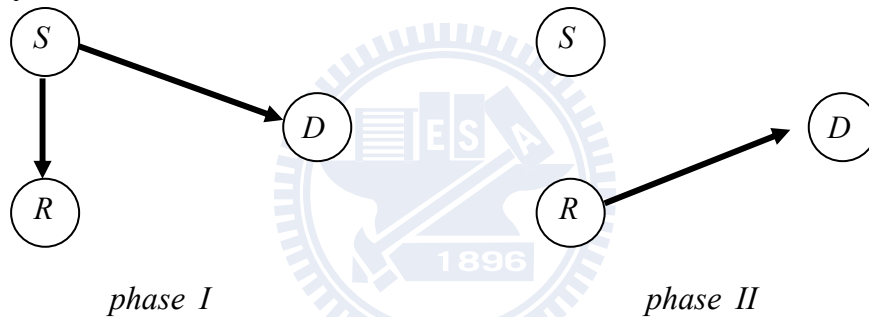


圖 5-1 AF protocol 示意圖

Phase 1 時，Source 傳送訊號給 Destination 端，Relay 端也幫忙聽 Source 所傳的訊號。Phase 2 時，Relay 端，僅把訊號做 normalized，再將訊號傳到 destination 端。接收端解碼時，將 phase1 與 phase2 接收到的訊號作適當的結合後，進行解碼的動作。AF 的壞處，就是會將 phase 1 中，Source 與 Relay 端間的雜訊做放大，影響接收端解碼的性能。

5.1.2 Decode and Forward

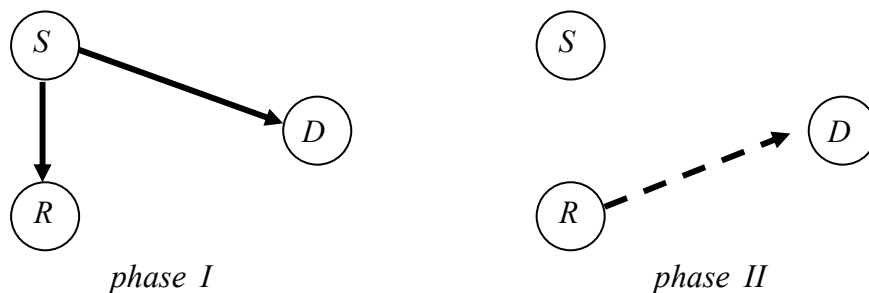


圖 5-2 DF protocol 示意圖

Phase 1 時，Source 傳送訊號給 Destination 端，Relay 端也幫忙聽 Source 所傳的訊號。假使 Relay 端能夠對 Source 所傳的訊號，成功解碼，Relay 會在 phase 2 時，對資訊重新編碼，再傳送訊號給 Destination 端；

反之，假如 Relay 端解碼失敗，在 phase 2 時，Relay 端不傳訊號。接收端解碼時，如同 AF protocol，將 phase1 與 phase2 接收到的訊號作適當的結合後，進行解碼的動作。

以下文章的探討，著重在於 decode-and-forward protocol 的應用，將 Alamouti code、space time trellis code 加入合作式通訊中，說明這些編碼在 MIMO 系統中，需要注意的準則與所提供的效能貢獻。

5.2 DF 合作式通訊—符碼(symbol)錯誤率分析

5.2.1 A simplified Cooperation model

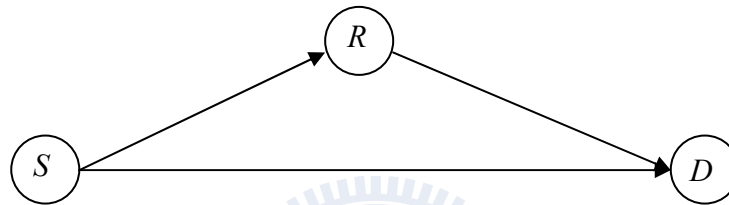


圖 5-3 A simplified Cooperation model 示意圖

依據 5.1 對於 DF protocol 的介紹，以數學式子表示此 system model，在 phase1 時，source 發射訊號，destination 與 relay 端聽取訊號：

$$y_{s,d} = \sqrt{P_1} h_{s,d} x + \eta_{s,d} \quad (\text{Eq. 5-1})$$

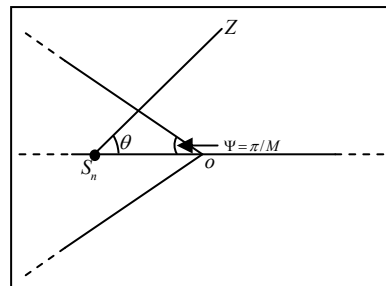
$$y_{s,r} = \sqrt{P_1} h_{s,r} x + \eta_{s,r} \quad (\text{Eq. 5-2})$$

在 phase2 時，relay 發射訊號，destination 端聽取訊號：

$$y_{r,d} = \sqrt{\tilde{P}_2} h_{r,d} x + \eta_{r,d} \quad (\tilde{P}_2 = 0 \text{ or } P_2) \quad (\text{Eq. 5-3})$$

假設 source、relay、destination 各自擁有單一根天線。 P_1 、 P_2 分別代表 source 與 relay 端的發射訊號能量， $h_{s,d}$ 、 $h_{s,r}$ 、 $h_{r,d}$ 代表 source-destination、source-relay 與 relay-destination 之間的 channel 係數，在此假設為 $CN(0,1)$ ； $\eta_{s,d}$ 、 $\eta_{s,r}$ 代表 AWGN，在此假設為 $CN(0, N_0)$ 。根據[16]，M-PSK 在 point-to-point 的傳收符碼錯誤率，表示如下：

$$\Psi_{PSK}(\rho) = \frac{1}{\pi} \int_0^{(M-1)\pi/M} \exp\left(-\frac{b_{PSK}\rho}{\sin^2 \theta}\right) d\theta \quad (\text{Eq. 5-4})$$



其中， $b_{PSK} = \sin^2(\pi/M)$ ， ρ 代表接收端的 SNR， ρ 又可表示為 $P_i|h|^2/N_0$ 。由於在此假設通道為 Rayleigh distribution，故可先對通道積分，獲得下列式子：

$$\Psi(\gamma) = \frac{1}{\pi} \int_0^{(M-1)\pi/M} \frac{1}{1 + \frac{b_{PSK} P_1 \delta_{i,j}^2}{N_0 \sin^2(\theta)}} d\theta = F_1\left(1 + \frac{b_{PSK} P_1 \delta_{i,j}^2}{N_0 \sin^2(\theta)}\right), \quad F_1(x(\theta)) = \frac{1}{\pi} \int_0^{(M-1)\pi/M} \frac{1}{x(\theta)} d\theta \quad (\text{Eq. 5-5})$$

應用上式於 decode-and-forward 合作式通訊中，(注意 relay 是否成功解調事件)

$$\begin{aligned} P_e &= P\{\text{relay decoding fail}\} \cdot P\{\text{destination decoding fail}\} + \\ &P\{\text{relay decoding successfully}\} \cdot P\{\text{destination decoding fail}\} \\ &= F_1\left(1 + \frac{b_{PSK} P_1 \delta_{s,d}^2}{N_0 \sin^2 \theta}\right) \cdot F_1\left(1 + \frac{b_{PSK} P_1 \delta_{s,r}^2}{N_0 \sin^2 \theta}\right) + \\ &F_1\left(\left(1 + \frac{b_{PSK} P_1 \delta_{s,d}^2}{N_0 \sin^2 \theta}\right) \cdot \left(1 + \frac{b_{PSK} P_2 \delta_{r,d}^2}{N_0 \sin^2 \theta}\right)\right) \cdot \left[1 - F_1\left(1 + \frac{b_{PSK} P_1 \delta_{s,r}^2}{N_0 \sin^2 \theta}\right)\right] \end{aligned} \quad (\text{Eq. 5-6})$$

假設 $\delta_{s,d} = \delta_{s,r} = \delta_{r,d}$ 、 $P_1 = P_2$ ，並將 $\sin^2 \theta = 1$ 帶入上式，得到錯誤率上界，

$$P_e \leq \frac{1}{(1+\gamma)^2} \cdot \left(\left(\frac{M-1}{M}\right)^2 + \frac{M-1}{M} - \left(\frac{M-1}{M}\right) \cdot \frac{1}{\gamma} \right) \quad \gamma = 1 + \frac{b_{PSK} P \delta^2}{N_0 \sin^2 \theta} \quad (\text{Eq. 5-7})$$

先觀察此 model 的 diversity 為 2，因為 F_1 積分不會對式子中 SNR 的參數有所影響。

5.2.2 Alamouti code in DF cooperation model

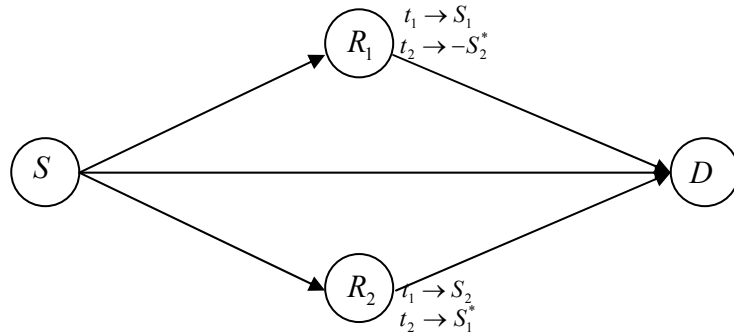


圖 5-4 Alamouti code in DF cooperation model 示意圖

同 5.2.1，我們以數學式子，表示此 system model。因為使用的是 Alamouti scheme，所以 terminal 端，每個 phase 都要發出兩個訊號(ex: S_1 、 S_2)。在 phase1 時，source 發出訊號，兩個 relay 與 destination 同時聽取訊號。

$$y_{s,d} = \sqrt{P_1} h_{s,d} x + \eta_{s,d} \quad (\text{Eq. 5-8})$$

$$y_{s,R_1} = \sqrt{P_1} h_{s,R_1} x + \eta_{s,R_1} \quad (\text{Eq. 5-9})$$

$$y_{s,R_2} = \sqrt{P_1} h_{s,R_2} x + \eta_{s,R_2} \quad (\text{Eq. 5-10})$$

假使 relay 端能夠對 phase1 的訊號解調成功，在 phase2 時，relay 端會重新編碼，使用 Alamouti scheme 的方法傳輸。

$$y_{R_1,d} = \sqrt{\tilde{P}_2} h_{R_1,d} x + \eta_{R_1,d} \quad (\text{Eq. 5-11})$$

$$y_{R_2,d} = \sqrt{\tilde{P}_2} h_{R_2,d} x + \eta_{R_2,d} \quad (\tilde{P}_2 = 0 \text{ or } P_2) \quad (\text{Eq. 5-12})$$

使用 Alamouti code，必須注意到一個特性，假如 phase1，source-relay 間的連結很差的話，一次 symbol 傳送的錯誤，會造成兩個 symbol 無法同時解調成功；所以在 source-relay 間的錯誤率必須提升兩倍：

$$F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,r}^2}{N_0 \sin^2 \theta} \right) \rightarrow 2 \cdot F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,r}^2}{N_0 \sin^2 \theta} \right) \quad (\text{Eq. 5-13})$$

此 system model 有三個產生錯誤的事件，敘述如下

1) 在 Phase1 時的三條連結皆產生錯誤，

$$\begin{aligned} \bar{P}_{e1} &= P \{D \text{ decoding fail}\} \times P \{R_1 \text{ decoding fail}\} \cdot P \{R_2 \text{ decoding fail}\} = \\ &F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,D}^2}{N_0 \sin^2 \theta} \right) \times 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_1}^2}{N_0 \sin^2 \theta} \right) \times 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_2}^2}{N_0 \sin^2 \theta} \right) \end{aligned} \quad (\text{Eq. 5-14})$$

2) 在 Phase1，其中一個 relay 解調錯誤，

$$\begin{aligned} \bar{P}_{e2} &= P \{R_1 \text{ decoding successfully}\} \cdot P \{R_2 \text{ decoding fail}\} + \\ &P \{R_2 \text{ decoding decoding fail}\} \cdot P \{R_1 \text{ decoding successfully}\} = \\ &\left[1 - 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_1}^2}{N_0 \sin^2 \theta} \right) \right] \times F_1 \left[\left(1 + \frac{b_{PSK} P_1 \delta_{s,D}^2}{N_0 \sin^2 \theta} \right) \left(1 + \frac{b_{PSK} P_2 \delta_{R_1,D}^2}{N_0 \sin^2 \theta} \right) \right] \times 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_2}^2}{N_0 \sin^2 \theta} \right) + 3 \\ &\left[1 - 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_2}^2}{N_0 \sin^2 \theta} \right) \right] \times F_1 \left[\left(1 + \frac{b_{PSK} P_1 \delta_{s,D}^2}{N_0 \sin^2 \theta} \right) \left(1 + \frac{b_{PSK} P_2 \delta_{R_2,D}^2}{N_0 \sin^2 \theta} \right) \right] \times 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_1}^2}{N_0 \sin^2 \theta} \right) \end{aligned} \quad (\text{Eq. 5-15})$$

3) 當兩個 relay 解調成功，卻在 destination 端解調錯誤

$$\begin{aligned} \bar{P}_{e3} &= P \{D \text{ decoding successfully} \mid R_1 \text{ decoding successfully}, R_2 \text{ decoding successfully}\} = \\ &F_1 \left[\left(1 + \frac{b_{PSK} P_1 \delta_{s,D}^2}{N_0 \sin^2 \theta} \right) \left(1 + \frac{b_{PSK} P_2 \delta_{R_1,D}^2}{N_0 \sin^2 \theta} \right) \left(1 + \frac{b_{PSK} P_2 \delta_{R_2,D}^2}{N_0 \sin^2 \theta} \right) \right] \times \\ &\left[1 - 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_2}^2}{N_0 \sin^2 \theta} \right) \right] \times \left[1 - 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_1}^2}{N_0 \sin^2 \theta} \right) \right] \end{aligned} \quad (\text{Eq. 5-16})$$

此 system model 的錯誤率，為上面三個事件的錯誤率相加：

$$\bar{P}_e = \bar{P}_{e1} + \bar{P}_{e2} + \bar{P}_{e3}$$

假設 $\delta_{s,d}$, $\delta_{s,r}$, δ_{rd} , $P_1=P_2$, 並將 $\sin^2 \theta=1$ 帶入上式, 得到錯誤率上界,

$$\bar{P}_e \leq \frac{1}{(1+\gamma)^3} \cdot \left(4\left(\frac{M-1}{M}\right)^3 + 4\left(\frac{M-1}{M}\right)^2 + \frac{M-1}{M} + \left(8\left(\frac{M-1}{M}\right)^3 + 4\left(\frac{M-1}{M}\right)^2 \right) / (1+\gamma) + 4\left(\frac{M-1}{M}\right)^3 / (1+\gamma)^2 \right) \quad (\text{Eq. 5-17})$$

$$\text{where } \gamma = 1 + \frac{b_{PSK} P \delta^2}{N_0 \sin^2 \theta}$$

先觀察此 model 的 diversity 為 3, 因為 F_1 積分不會對式子中 SNR 的參數有所影響。

5.2.3 Alamouti code in DF cooperation model without source destination link

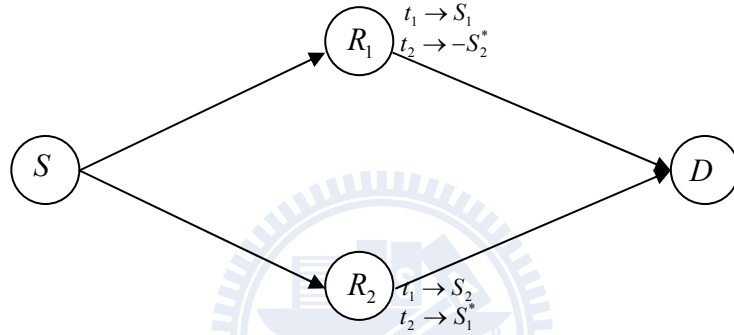


圖 5-5 Alamouti code in DF cooperation model without source destination link 示意圖

與 5.2.2 不同的地方在於, Phase1 時, destination 端不聽取訊號, 其他的部分皆與 5.2.2 相同。System model 即參考 5.2.2, 以下作錯誤率分析。考慮三個事件:

- 1) 在 Phase1 時的兩條連結皆產生錯誤

$$\hat{P}_{e1} = P\{R_1 \text{ decoding fail}\} \cdot P\{R_2 \text{ decoding fail}\} = 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_1}^2}{N_0 \sin^2 \theta} \right) \times 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_2}^2}{N_0 \sin^2 \theta} \right) \quad (\text{Eq. 5-18})$$

- 2) 在 Phase1 時, 其中有一個 relay 解調錯誤,

$$\begin{aligned} \hat{P}_{e2} &= P\{R_1 \text{ decoding successfully}\} \cdot P\{R_2 \text{ decoding fail}\} + \\ &P\{R_2 \text{ decoding decoding fail}\} \cdot P\{R_1 \text{ decoding successfully}\} = \\ &\left[1 - 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_1}^2}{N_0 \sin^2 \theta} \right) \right] \times 2F_1 \left[\left(1 + \frac{b_{PSK} P_2 \delta_{R_1,D}^2}{N_0 \sin^2 \theta} \right) \right] \times 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_2}^2}{N_0 \sin^2 \theta} \right) + \\ &\left[1 - 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_2}^2}{N_0 \sin^2 \theta} \right) \right] \times 2F_1 \left[\left(1 + \frac{b_{PSK} P_2 \delta_{R_2,D}^2}{N_0 \sin^2 \theta} \right) \right] \times 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_1}^2}{N_0 \sin^2 \theta} \right) \end{aligned} \quad (\text{Eq. 5-19})$$

- 3) 當兩個 relay 解調成功, 卻在 destination 端解調錯誤,

$$\hat{P}_{e3} = P\{D \text{ decoding successfully} \mid R_1 \text{ decoding successfully}, R_2 \text{ decoding successfully}\} =$$

$$F_1 \left[\left(1 + \frac{b_{PSK} P_2 \delta_{R_1,D}^2}{N_0 \sin^2 \theta} \right) \left(1 + \frac{b_{PSK} P_2 \delta_{R_2,D}^2}{N_0 \sin^2 \theta} \right) \right] \times \left[1 - 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_2}^2}{N_0 \sin^2 \theta} \right) \right] \times \left[1 - 2F_1 \left(1 + \frac{b_{PSK} P_1 \delta_{s,R_2}^2}{N_0 \sin^2 \theta} \right) \right]$$

(Eq. 5-20)

此 system model 的錯誤率，為上面三個事件的錯誤率相加：

$$\hat{P}_e = \hat{P}_{e1} + \hat{P}_{e2} + \hat{P}_{e3}$$

假設 $\delta_{s,d} = \delta_{s,r} = \delta_{r,d}$ 、 $P_1 = P_2$ ，並將 $\sin^2 \theta = 1$ 帶入上式，得到錯誤率上界，

$$\hat{P}_e \leq \frac{1}{(1+\gamma)^2} \cdot \left(12 \left(\frac{M-1}{M} \right)^2 + \frac{M-1}{M} - 20 \left(\frac{M-1}{M} \right)^2 / (1+\gamma) + 4 \left(\frac{M-1}{M} \right)^3 / (1+\gamma)^2 \right) \quad (\text{Eq. 5-21})$$

$$\text{where } \gamma = 1 + \frac{b_{PSK} P \delta^2}{N_0 \sin^2 \theta}$$

先觀察，此 model 的 diversity 為 2，因為 F_1 積分不會對式子中 SNR 的參數有所影響。觀察上面三個 model，比較它們間的符碼錯誤率上限為 $\hat{P}_e < \bar{P}_e < P_e$ 。以下是對於上面三個 model 所作的模擬，總共模擬 $1e7$ 個符碼個數(symbol)

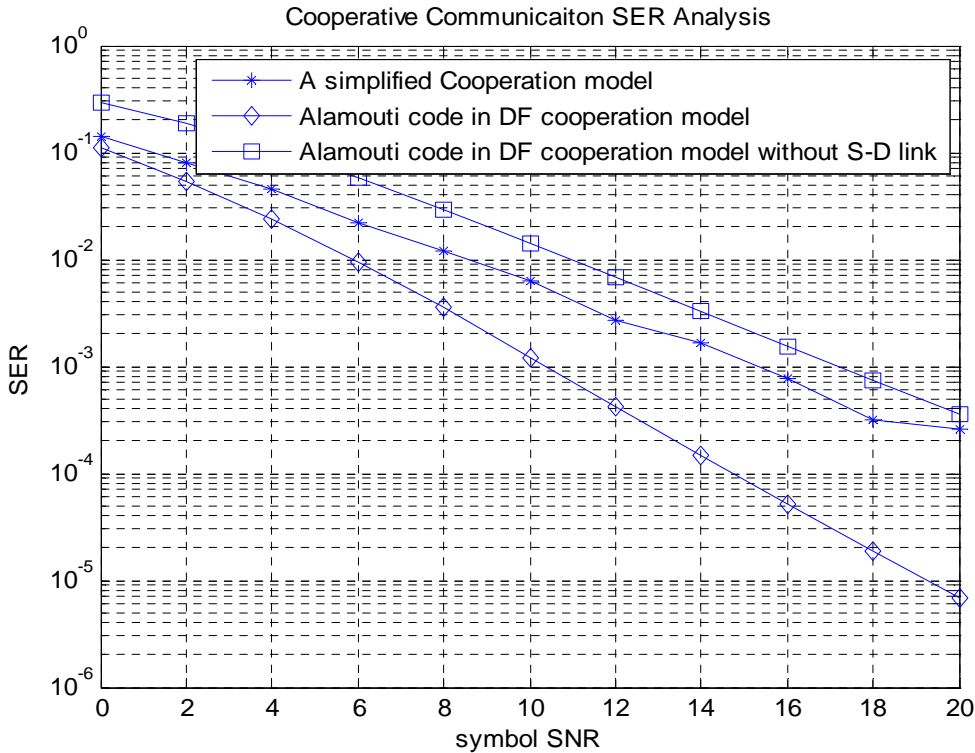


圖 5-6 不同合作式通訊 protocol 在符碼錯誤率的表現比較

5.3 DF 合作式通訊—封包錯誤率分析

本章節討論空時編碼的封包錯誤率，將籬柵空時編碼加入合作式通訊，先討論編碼設計的準則，再對一些合作式通訊方法做實驗。

5.3.1 應用於 DF 合作式通訊上的籬柵空時編碼

為了完整將籬柵空時編碼載入 DF 合作式通訊中，每次傳送一組 information bits，使用“2L” time slots(假設封包長度為 L)。

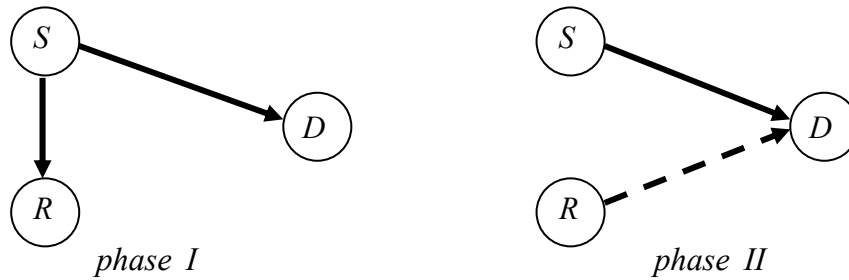
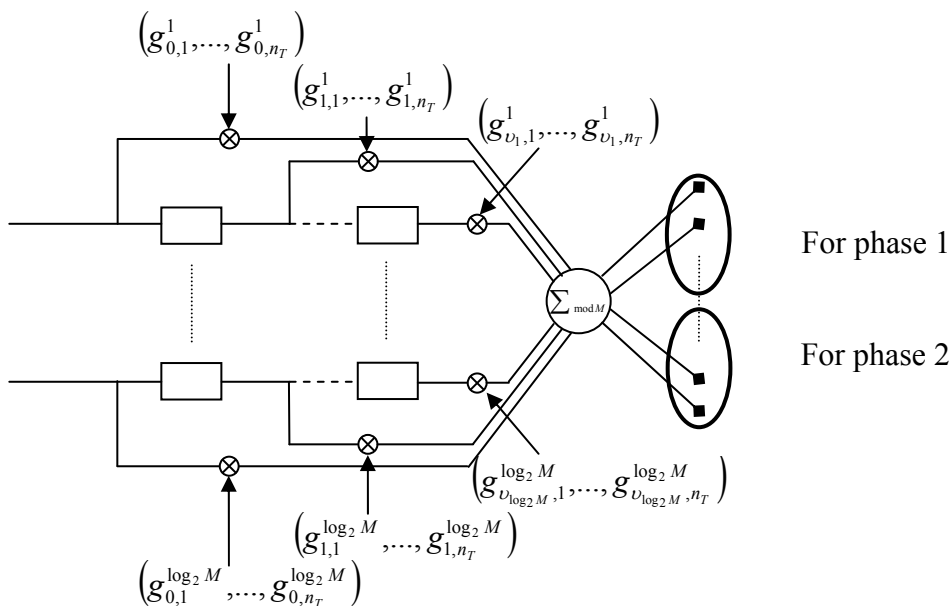


圖 5-7 DF protocol with S-D link in phase 2 示意圖

Phase 1 時，Source 傳送資料給 Destination 端，Relay 端也幫忙聽 Source 所傳的 signal。Phase 2 時，假使 source-relay 間的通道很好，使得 Relay 端能夠對 Source 所傳的訊號，皆能成功解碼，然後再傳送訊號給 Destination 端。

5.3.2 分散式籬柵空時碼編碼結構

Source 端的籬柵空時碼編碼結構，先以最複雜的例子來說明。Source 端傳送訊號時，分成兩個部分，phase 1 和 phase 2。因此將籬柵空時碼編碼器所製造出來的訊號，分成兩部分傳送。設計 Source 端的籬柵空時碼編碼器時，假設天線個數為 $2n_s$ 根(實際上只有 n_s 根)。Phase 1 時，傳送第 $1 \sim n_s$ 根天線上的 symbols；phase 2 時，傳送第 $n_s+1 \sim 2n_s$ 根天線上的 symbols。



Relay 端的籬柵空時碼 encoder structure, 由於只需使用到一個 frame 的時間;”有別”於 Source 端需要用到 2 個 frame 的時間發送訊號, 所以 Relay 端用傳統的籬柵空時碼編碼器結構即可, 不需將傳送天線分成兩個部分。

5.3.3 定義分散式籬柵空時碼矩陣

就像討論集中式籬柵空時碼般, 我們會想製造一個 difference matrix 來討論分散式籬柵空時碼編碼的方法, 讓籬柵空時碼用於合作式通訊時, 找到最佳 {g}, 使得編碼效果最好。這些 symbol matrix 所訂出來的方式, 是為了用於運算上的方便, 傳統上輸出輸入的矩陣關係為 $\mathbf{R}=\mathbf{H}\mathbf{X}+\mathbf{N}$ 。根據 Decode-and-Forward 的觀念。使用分散式籬柵空時碼時, 在 phase 1 時, Source 端會先發出一段訊號, 訊號的長度為一個封包長度, 將此段訊號, 用矩陣表示。

$$\mathbf{X}_1 = \begin{bmatrix} x_1^1 & \cdots & x_{t_1}^1 & \cdots & x_L^1 \\ \vdots & \ddots & \cdots & \ddots & \vdots \\ x_1^{n_S} & \cdots & x_{t_1}^{n_S} & \cdots & x_L^{n_S} \end{bmatrix}_{n_S \times L}$$

在 phase 2 時, Source 端再發出一段訊號, 訊號長度也是一個 frame size。此段訊號, 用矩陣表示。

$$\mathbf{X}_2 = \begin{bmatrix} x_{L+1}^1 & \cdots & x_{t_2}^1 & \cdots & x_{2L}^1 \\ \vdots & \ddots & \cdots & \ddots & \vdots \\ x_{L+1}^{n_S} & \cdots & x_{t_2}^{n_S} & \cdots & x_{2L}^{n_S} \end{bmatrix}_{n_S \times L}$$

在 phase 2 時, Relay 端發出訊號。訊號的長度為一個 frame size, 此段訊號, 用矩陣表示。

$$\mathbf{R} = \begin{bmatrix} r_{L+1}^1 & \cdots & r_{t_2}^1 & \cdots & r_{2L}^1 \\ \vdots & \ddots & \cdots & \ddots & \vdots \\ r_{L+1}^{n_R} & \cdots & r_{t_2}^{n_R} & \cdots & r_{2L}^{n_R} \end{bmatrix}_{n_R \times L}$$

在兩個封包時間長度內, 對於所有的 code symbol sequences, 所產生的分散式籬柵空時碼矩陣, 表示如下:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 \\ \mathbf{0} & \mathbf{R} \end{bmatrix}_{(n_S+n_R) \times 2L}$$

將各種分散式籬柵空時碼所會用到的 Difference 矩陣表示如下:

$$\mathbf{B}(\mathbf{X}, \hat{\mathbf{X}}) = \mathbf{X} - \hat{\mathbf{X}} = \begin{bmatrix} x_1^1 - \hat{x}_1^1 & \cdots & x_L^1 - \hat{x}_L^1 & x_{L+1}^1 - \hat{x}_{L+1}^1 & \cdots & x_{2L}^1 - \hat{x}_{2L}^1 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_1^{n_S} - \hat{x}_1^{n_S} & \cdots & x_L^{n_S} - \hat{x}_L^{n_S} & x_{L+1}^{n_S} - \hat{x}_{L+1}^{n_S} & \cdots & x_{2L}^{n_S} - \hat{x}_{2L}^{n_S} \\ 0 & \cdots & 0 & r_{L+1}^1 - \hat{r}_{L+1}^1 & \cdots & r_{2L}^1 - \hat{r}_{2L}^1 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & r_{L+1}^{n_R} - \hat{r}_{L+1}^{n_R} & \cdots & r_{2L}^{n_R} - \hat{r}_{2L}^{n_R} \end{bmatrix}_{(n_S+n_R) \times n_D}$$

$$\mathbf{Q}(\mathbf{X}, \hat{\mathbf{X}}) = \mathbf{X}_1 - \hat{\mathbf{X}}_1 = \begin{bmatrix} x_1^1 - \hat{x}_1^1 & \cdots & x_L^1 - \hat{x}_L^1 \\ \vdots & \ddots & \vdots \\ x_1^{n_S} - \hat{x}_1^{n_S} & \cdots & x_L^{n_S} - \hat{x}_L^{n_S} \end{bmatrix}_{n_S \times L}$$

$$\mathbf{C}(\mathbf{X}, \hat{\mathbf{X}}) = [\mathbf{X}_2 - \hat{\mathbf{X}}_2 \quad \mathbf{R} - \hat{\mathbf{R}}]^T = \begin{bmatrix} x_{L+1}^1 & \cdots & x_{2L}^1 \\ \vdots & \ddots & \vdots \\ x_{L+1}^{n_S} & \cdots & x_{2L}^{n_S} \\ x_{L+1}^1 & \cdots & x_{2L}^1 \\ \vdots & \ddots & \vdots \\ x_{L+1}^{n_R} & \cdots & x_{2L}^{n_R} \end{bmatrix}_{(n_S+n_R) \times L}$$

5.3.4 接收端的解碼器

由於合作式通訊分成兩個 phase 傳送 signal，所以接收端也會分別在兩個 phase，接收到訊號；將兩個 phase 所收到的訊號，用矩陣表示如下：

$$\mathbf{Y} = [\mathbf{Y}_1 \quad \mathbf{Y}_2] = \begin{bmatrix} y_1^1 & \cdots & y_{t_1}^1 & \cdots & y_L^1 & y_{L+1}^1 & \cdots & y_{t_2}^1 & \cdots & y_{2L}^1 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ y_1^{n_D} & \cdots & y_{t_1}^{n_D} & \cdots & y_L^{n_D} & y_{L+1}^{n_D} & \cdots & y_{t_2}^{n_D} & \cdots & y_{2L}^{n_D} \end{bmatrix}_{n_D \times 2L}$$

← For phase 1
← For phase 2

合作式通訊使用籬柵空時碼來做訊號編碼，所以解碼時，使用 Viterbi algorithm 解碼。不過在計算 branch metric 時，需要考慮兩個 frame 的總影響！注意 ML decision 的方法如下：

$$\text{Branch Metric} = \sum_{t_1=1}^L |\mathbf{Y}_{t_1} - \mathbf{H}_{SD,t_1} \mathbf{X}_{t_1}|^2 + \sum_{t_1=L+1}^{2L} |\mathbf{Y}_{t_1} - \mathbf{H}_{SD,t_1} \mathbf{X}_{t_1} - \mathbf{H}_{RD,t_1} \mathbf{R}_{t_1}|^2 \quad (\text{Eq. 5-22})$$

5.3.5 慢速衰減通道下的設計準則

Slow-Fading 狀況下，Source 與 Destination 間的 channel state，在 Phase 1 與 Phase 2 間不會改變。 $\mathbf{H}_{SD,t_1} = \mathbf{H}_{SD,t_2}$ 。仿照第二章的導證過程，使用 Viterbi ML 解碼演算法，

將通道的影響積分過後，得到的錯誤率如下：

$$P(\mathbf{X}, \hat{\mathbf{X}}) \leq \left(\prod_{l=1}^{r_s} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} \leq \left(\prod_{l=1}^{r_s} \frac{1}{\alpha_l} \right)^{n_D} \left(\frac{E_s}{4N_0} \right)^{-r_s n_D} \quad \begin{array}{l} \alpha_l : \text{positive eigen value of } \mathbf{A} \\ r_s : \text{rank of } \mathbf{A} \end{array} \quad (\text{Eq. 5-23})$$

由 FER 看來，在 slow-fading 狀況下時，遵守集中式籬柵空時碼編碼器的設計方法。

- \mathbf{A} 的 rank 愈高愈好， \mathbf{A} 的 rank 決定分散式籬柵空時碼的 transmit diversity。 $(\mathbf{A}=\mathbf{B}\mathbf{B}^H)$
- \mathbf{A} 的 determinant 愈高愈好， \mathbf{A} 的 determinant 決定分散式籬柵空時碼的 coding gain。

以下是針對 Slow-Fading 通道下，模擬不同籬柵空時編碼的效能，Slow-fading 狀況下，不同 memory 個數，對於錯誤率的影響。上為模擬為 $n_S=n_R=n_D=1$ ，對於每個分散式籬柵空時碼都具有相同的 rank of \mathbf{A} (transmit diversity=2)，但不同的分散式籬柵空時碼間，存在 states 個數的差異，states 個數愈多，愈能增 coding gain。此外，我們也可以觀察 determinant，對於 coding gain 的影響，determinant 愈大，coding 就愈大。

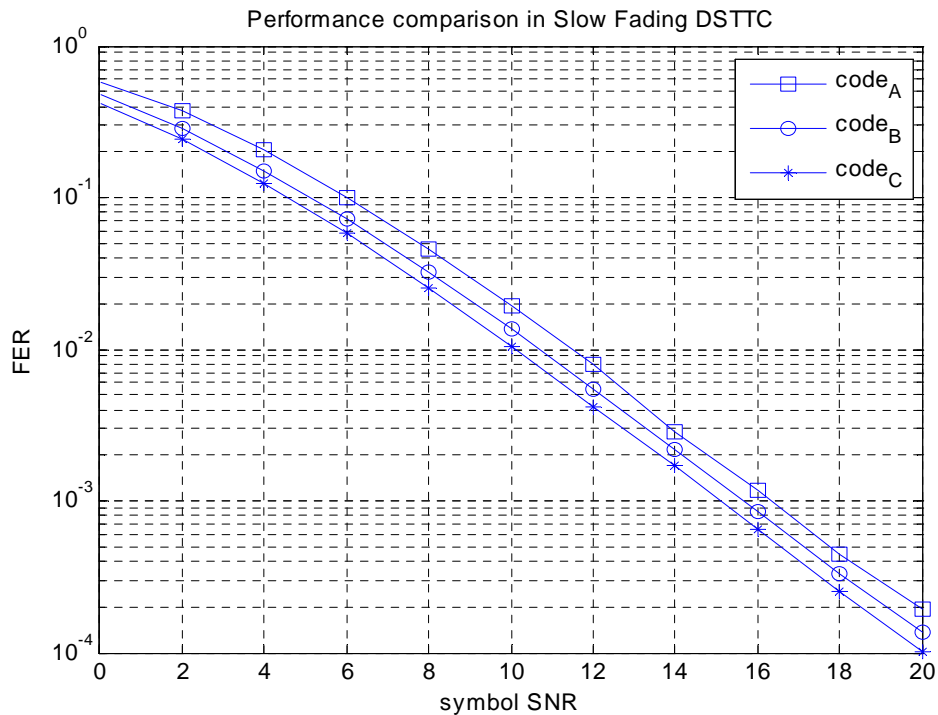


圖 5-8 determinant criterion 應用於分散式籬柵空時碼中(for slow-fading channel)

分散式 籬柵空時碼 states	Generator sequence (Source)	Generator sequence (Relay)	rank	determinant
Code A 4-states	$g^1=[(00),(12)]$ $g^2=[(20),(20)]$	$g^1=[22]$ $g^2=[11]$	2	32
Code B 8-states	$g^1=[(22),(20)]$ $g^2=[(00),(12),(02)]$	$g^1=[12]$ $g^2=[202]$	2	60
Code C	$g^1=[(20),(20),(22)]$	$g^1=[102]$	2	80

16-states	$g^2=[(02),(10),(12)]$	$g^2=[221]$		
-----------	------------------------	-------------	--	--

5.3.6 類慢速衰減通道下的設計準則

Quasi-slow Fading 狀況下，Source 與 Destination 間的 channel state，在 Phase 1 與 Phase 2 間會改變。 $\mathbf{H}_{SD,t_1} \neq \mathbf{H}_{SD,t_2}$ ，使用 Viterbi ML decoder 的錯誤率如下：

$$P(\mathbf{X}, \hat{\mathbf{X}}) \leq 2 \left(\prod_{m=1}^{r_{q1}} \frac{1}{1 + \frac{E_s}{4N_0} \beta_m} \prod_{n=1}^{r_{q2}} \frac{1}{1 + \frac{E_s}{4N_0} \lambda_n} \right)^{n_D} \approx 2 \left(\prod_{l=1}^{r_{q1}} \frac{1}{\beta_m} \right)^{n_D} \left(\prod_{n=1}^{r_{q2}} \frac{1}{\lambda_n} \right)^{n_D} \left(\frac{E_s}{4N_0} \right)^{-n_D(r_{q1}+r_{q2})} \quad (\text{Eq. 5-24})$$

β_m : positive eigen value of \mathbf{D} λ_n : positive eigen value of \mathbf{G}
 r_{q1} : rank of \mathbf{D} r_{q2} : rank of \mathbf{G}

由 FEP 看來，在 quasi-slow fading 狀況下時，encoder 的設計方法。

- \mathbf{D} 與 \mathbf{G} 的 rank 愈高愈好， \mathbf{D} 與 \mathbf{G} 的 rank 決定分散式籬柵空時碼的 transmit diversity。 ($\mathbf{D}=\mathbf{Q}\mathbf{Q}^H$ 、 $\mathbf{G}=\mathbf{C}\mathbf{C}^H$)
- \mathbf{D} 與 \mathbf{G} 的 determinant 愈高愈好， \mathbf{D} 與 \mathbf{G} 的 determinant 決定分散式籬柵空時碼的 coding gain

當 Source 與 Relay 的傳送天線，皆為一根時， $n_S=n_R=1$ ：

$$P(\mathbf{X}, \hat{\mathbf{X}}) \leq \frac{2}{d_E^2} \prod_{n=1}^{r_{q2}} \frac{1}{\lambda_n} \left(\frac{E_s}{4N_0} \right)^{-(1+r_{q2})} \quad \text{where } d_E^2 = \sum_{t=1}^L |x_t^i - \hat{x}_t^i|^2 \quad (\text{Eq. 5-25})$$

所有的特性(diversity and coding gain)，皆與 \mathbf{D} 矩陣無關。定義 $d_E^2 \prod_{n=1}^{r_{q2}} \lambda_n$ 為 product coding gain(PCG)。如此一來，分散式籬柵空時碼的 transmit diversity 純粹交由 \mathbf{G} 矩陣決定，coding gain 交由 PCG 決定。

以下是針對 Quasi Slow Fading 通道下，模擬不同籬柵空時編碼的效能，Quasi-Slow fading 狀況下，不同 memory 個數，對於錯誤率的影響。上為模擬為 $n_S=n_R=n_D=1$ ，對於每個分散式籬柵空時碼都具有相同的 rank of \mathbf{A} (transmit diversity=2)，但不同的分散式籬柵空時碼間，存在 states 個數的差異，states 個數愈多，愈能增 coding gain。此外，我們也可以觀察 determinant，對於 coding gain 的影響，PCG 愈大，coding gain 就愈大。

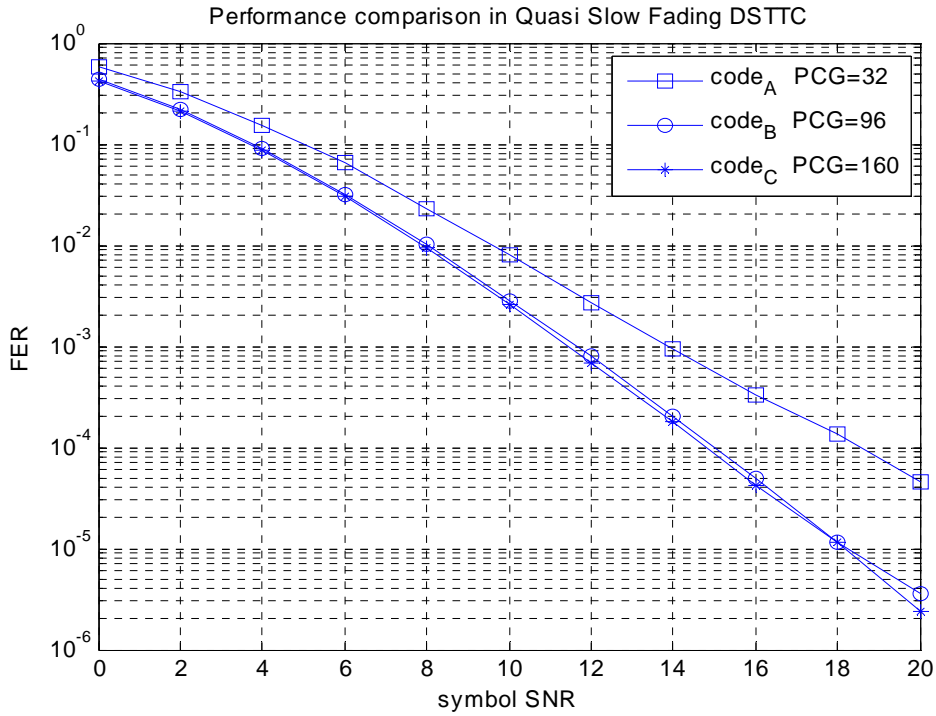


圖 5-9 determinant criterion 應用於分散式籬柵空時碼中(for Quasi slow-fading channel)

分散式 籬柵空時碼 states	Generator Sequence (Source)	Generator Sequence (Relay)	rank	determinant	d_E^2	PCG($\det \cdot d_E^2$)
Code A 8-states	$g^1=[(10),(32)]$ $g^2=[(12),(03),(00)]$	$g^1=[23]$ $g^2=[201]$	2	16	2	32
Code B 8-states	$g^1=[(22),(00)]$ $g^2=[(10),(13),(12)]$	$g^1=[12]$ $g^2=[230]$	2	16	6	96
Code C 8-states	$g^1=[(12),(10)]$ $g^2=[(20),(11),(22)]$	$g^1=[12]$ $g^2=[310]$	2	16	10	160

5.3.7 比較慢速與類慢速衰減通道狀況

比較 slow-fading 與 quasi-slow fading 狀況，transmit diversity 最大分別為 $(n_S+n_R) \cdot n_D$ 與 $(2n_S+n_R) \cdot n_D$ 。當 $n_S=n_R=n_D=1$ 時，slow-fading 分散式籬柵空時碼 最大 transmit diversity=2，quasi-slow fading 分散式籬柵空時碼 最大 transmit diversity=3。

5.4 應用於合作式通訊上的籬柵空時編碼實驗

5.4.1 Hard/Soft decision 解碼比較

參考 5.2.1 節的 system model，將籬柵空時編碼應用於其中；此部分與 5.3 節不同的地方，在於 phase2 時，source 端不再發射訊號。根據 5.3 節的導證，假設 source 與 relay 端的編碼方式一樣，天線個數皆為一根，在合作式通訊中至少可以達到 diversity=2 的傳

輸。模擬過程中，實驗 $1e7$ 次，每次實驗都傳一個封包，封包內含 260bits，比較硬式解碼與軟式解碼的差別。

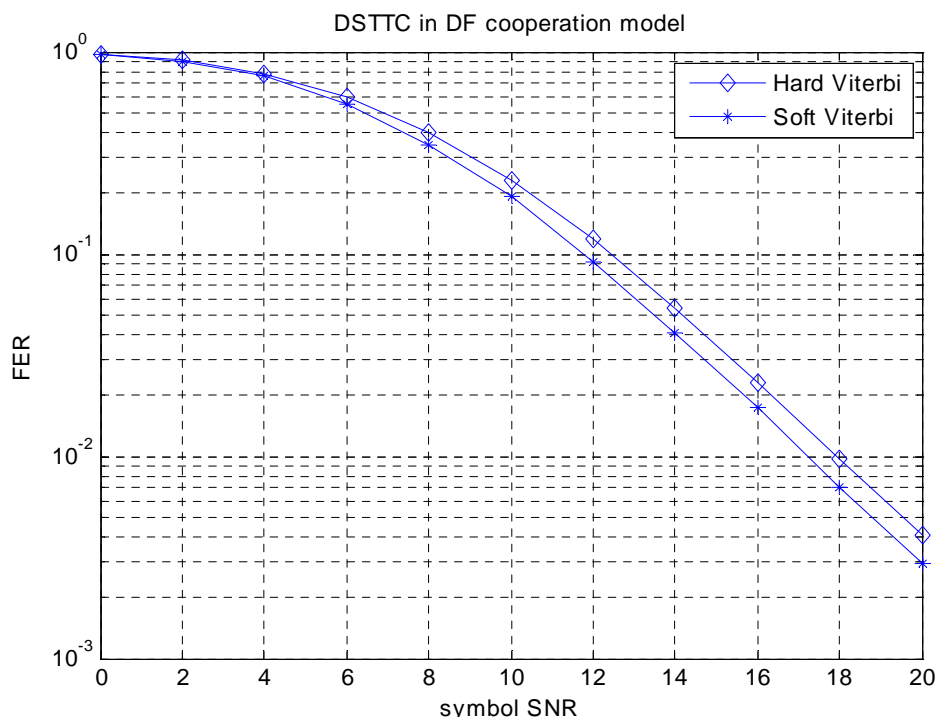


圖 5-10 Hard v.s. Soft decision 比較

分散式籬柵空時碼 states	Generator sequence (Source)	Generator sequence (Relay)	rank	determinant
4-states	$g^1=[(00),(20)]$ $g^2=[(00),(10)]$	$g^1=[02]$ $g^2=[01]$	2	4

如同觀念，軟式解碼較硬式解碼方法好。

5.4.2 不同 system model 效能上的比較

針對 5.2 的三種 system model，我們加入空時籬柵碼於其中，從 frame error probability，探討分散式籬柵空時碼的設計方法。假設通道的機率分佈為 rayleigh distribution，在一個封包長度時間內，通道不會改變。

5.4.2.1 A simplified Cooperation model

參考 5.2.1 的架構圖，假如 relay 解碼成功，接收端處理兩個相異 coded symbol，在星狀圖上的距離，用下列式子表示：

$$\sum_{j=1}^{n_D} \left(\sum_{t_1=1}^L \sum_{i_1=1}^{n_S} h_{SD}^{j,i_1} (x_{t_1}^{i_1} - \hat{x}_{t_1}^{i_1}) \right)^2 + \sum_{t_2=L+1}^{2L} \sum_{i_2=1}^{n_R} h_{RD}^{j,i_2} (x_{t_2}^{i_2} - \hat{x}_{t_2}^{i_2}) \right)^2 \quad (\text{Eq. 5-26})$$

假如 relay 端解碼失敗，接收端處理兩個相異 coded symbol，在星狀圖上的距離，用下

列式子表示：

$$\sum_{j=1}^{n_D} \sum_{t_1=1}^L \left| \sum_{i_1=1}^{n_S} h_{SD}^{j,i_1} (x_{t_1}^{i_1} - \hat{x}_{t_1}^{i_1}) \right|^2 \quad (\text{Eq. 5-27})$$

此外，relay 端處理兩相異 coded symbol，在星狀圖上的距離，用下列式子表示：

$$\sum_{j=1}^{n_R} \sum_{t_1=1}^L \left| \sum_{i_1=1}^{n_S} h_{SR}^{j,i_1} (x_{t_1}^{i_1} - \hat{x}_{t_1}^{i_1}) \right|^2 \quad (\text{Eq. 5-28})$$

根據[9]，假如 source 端的 space time code 具有 full transmit diversity，那麼空時碼的 exact FER 可以用下列式子表示：

$$P(\mathbf{X}, \hat{\mathbf{X}}) = \frac{1}{\pi} \int_0^{\pi/2} \prod_{i=1}^{n_S} \left[1 + \frac{E_s}{4N_0 \sin^2 \theta} \sum_{t=1}^L |x_t^i - \hat{x}_t^i|^2 \right]^{-n_R} d\theta \quad (\text{Eq. 5-29})$$

所以此 system 的錯誤率表現，可以表示如下

$$\sum_{j=1}^{n_D} \left(\sum_{t_1=1}^L \left| \sum_{i_1=1}^{n_S} h_{SD}^{j,i_1} (x_{t_1}^{i_1} - \hat{x}_{t_1}^{i_1}) \right|^2 + \sum_{t_2=L+1}^{2L} \left| \sum_{i_{R1}=1}^{n_{R1}} h_{R1D}^{j,i_1} (x_{t_2}^{i_{R1}} - \hat{x}_{t_2}^{i_{R1}}) + \sum_{i_{R2}=1}^{n_{R2}} h_{R2D}^{j,i_1} (x_{t_2}^{i_{R2}} - \hat{x}_{t_2}^{i_{R2}}) \right|^2 \right) \quad (\text{Eq. 5-30})$$

$$P_e = P\{\text{relay decoding fail}\} \cdot P\{\text{destination decoding fail}\} + P\{\text{relay decoding successfully}\} \cdot P\{\text{destination decoding fail}\}$$

$$\begin{aligned} &\leq P(\mathbf{X}, \hat{\mathbf{X}}) \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} + (1 - P(\mathbf{X}, \hat{\mathbf{X}})) \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} \cdot \left(\prod_{m=1}^{n_R} \left(1 + \frac{E_s}{4N_0} \beta_m \right) \right)^{-n_D} \\ &\leq \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_R} \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} + \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} \cdot \left(\prod_{m=1}^{n_R} \left(1 + \frac{E_s}{4N_0} \beta_m \right) \right)^{-n_D} \\ &\approx \left(\prod_{l=1}^{n_S} \frac{1}{\alpha_l} \right)^{n_R} \left(\frac{E_s}{4N_0} \right)^{-n_S n_R} \cdot \left(\prod_{l=1}^{n_S} \frac{1}{\alpha_l} \right)^{n_R} \left(\frac{E_s}{4N_0} \right)^{-n_S n_D} + \left(\prod_{l=1}^{n_S} \frac{1}{\alpha_l} \right)^{n_R} \left(\frac{E_s}{4N_0} \right)^{-n_S n_R} \cdot \left(\prod_{m=1}^{n_R} \frac{1}{\beta_m} \right)^{n_D} \left(\frac{E_s}{4N_0} \right)^{-n_R n_D} \end{aligned} \quad (\text{Eq. 5-31})$$

由上述的推導可知，假如 $n_{R1}=n_{R2}=n_R$ 、此系統的 diversity 為 $\min\{(n_R+n_D) \cdot n_S, (n_S+n_R) \cdot n_D\}$ 。

5.4.2.2 DSTTC in DF cooperation model

參考 5.2.2 的架構圖，假如 relay 解碼成功，接收端處理兩個相異 coded symbol，在星狀圖上的距離，用下列式子表示：

$$\sum_{j=1}^{n_D} \left(\sum_{t_1=1}^L \left| \sum_{i_1=1}^{n_S} h_{SD}^{j,i_1} (x_{t_1}^{i_1} - \hat{x}_{t_1}^{i_1}) \right|^2 + \sum_{t_2=L+1}^{2L} \left| \sum_{i_{R1}=1}^{n_{R1}} h_{R1D}^{j,i_1} (x_{t_2}^{i_{R1}} - \hat{x}_{t_2}^{i_{R1}}) + \sum_{i_{R2}=1}^{n_{R2}} h_{R2D}^{j,i_1} (x_{t_2}^{i_{R2}} - \hat{x}_{t_2}^{i_{R2}}) \right|^2 \right) \quad (\text{Eq. 5-32})$$

假如其中一個 relay 端解碼失敗，接收端處理兩個相異 coded symbol，在星狀圖上的距離，用下列式子表示：

$$\sum_{j=1}^{n_D} \left(\sum_{t_1=1}^L \left| \sum_{i_1=1}^{n_S} h_{SD}^{j,i_1} (x_{t_1}^{i_1} - \hat{x}_{t_1}^{i_1}) \right|^2 + \sum_{t_2=1}^{2L} \left| \sum_{i_{R1}=1}^{n_{R1}} h_{R1D}^{j,i_1} (x_{t_2}^{i_{R1}} - \hat{x}_{t_2}^{i_{R1}}) \right|^2 \right) \quad \text{or} \quad (\text{Eq. 5-33})$$

$$\sum_{j=1}^{n_D} \left(\sum_{t_1=1}^L \left| \sum_{i_1=1}^{n_S} h_{SD}^{j,i_1} (x_{t_1}^{i_1} - \hat{x}_{t_1}^{i_1}) \right|^2 + \sum_{t_2=L+1}^{2L} \left| \sum_{i_{R2}=1}^{n_{R2}} h_{R2D}^{j,i_1} (x_{t_2}^{i_{R2}} - \hat{x}_{t_2}^{i_{R2}}) \right|^2 \right) \quad (\text{Eq. 5-34})$$

假如兩個 Relay 端同時解碼失敗，接收端處理兩個相異 coded symbol，在星狀圖上的距離，用下列式子表示：

$$\sum_{j=1}^{n_D} \sum_{t_1=1}^L \left| \sum_{i_1=1}^{n_S} h_{SD}^{j,i_1} (x_{t_1}^{i_1} - \hat{x}_{t_1}^{i_1}) \right|^2 \quad (\text{Eq. 5-35})$$

同樣根據[9]，利用 Source 與 Relay 間的 exact 錯誤率，把此系統的錯誤率列於下方：

$$P_e = P\{\text{both relays decoding fail}\} \cdot P\{\text{destination decoding fail}\} + P\{\text{one relay decoding fail}\} \cdot P\{\text{destination decoding fail}\} + P\{\text{relays decoding successfully}\} \cdot P\{\text{destination decoding fail}\}$$

$$\begin{aligned} &\leq P(\mathbf{X}, \hat{\mathbf{X}}) \cdot P(\mathbf{X}, \hat{\mathbf{X}}) \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} + \\ &P(\mathbf{X}, \hat{\mathbf{X}}) \cdot (1 - P(\mathbf{X}, \hat{\mathbf{X}})) \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} \cdot \left(\prod_{m=1}^{n_{R1}} \left(1 + \frac{E_s}{4N_0} \beta_m \right) \right)^{-n_D} + \\ &P(\mathbf{X}, \hat{\mathbf{X}}) \cdot (1 - P(\mathbf{X}, \hat{\mathbf{X}})) \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} \cdot \left(\prod_{n=1}^{n_{R2}} \left(1 + \frac{E_s}{4N_0} \beta_n \right) \right)^{-n_D} + \\ &(1 - P(\mathbf{X}, \hat{\mathbf{X}})) \cdot (1 - P(\mathbf{X}, \hat{\mathbf{X}})) \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} \cdot \left(\prod_{m=1}^{r_{q2}} \left(1 + \frac{E_s}{4N_0} \beta_m \right) \right)^{-n_D} \\ &\leq \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_{R1}} \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_{R2}} \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} + \\ &\left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_{R2}} \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} \cdot \left(\prod_{m=1}^{n_{R1}} \left(1 + \frac{E_s}{4N_0} \beta_m \right) \right)^{-n_D} + \\ &\left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_{R1}} \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} \cdot \left(\prod_{n=1}^{n_{R2}} \left(1 + \frac{E_s}{4N_0} \beta_n \right) \right)^{-n_D} + \\ &\left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_D} \cdot \left(\prod_{m=1}^{r_{q2}} \left(1 + \frac{E_s}{4N_0} \beta_m \right) \right)^{-n_D} \end{aligned}$$

$$(\text{Eq. 5-36})$$

由上述的推導可知，假如 $n_{R1}=n_{R2}=n_R$ 、此系統的 diversity 為 $\min\{(2n_R+n_D)n_S, n_S n_R+(n_S+n_R)n_D, (n_S+2n_R)n_D\}$

5.4.2.3 DSTTC in DF cooperation model without S-D

參考 5.2.3 架構圖，假如 relay 解碼成功，接收端處理兩個相異 coded symbol，在星狀圖上的距離，用下列式子表示：

$$\sum_{j=1}^{n_D} \sum_{t_2=L+1}^{2L} \left| \sum_{i_{R1}=1}^{n_{R1}} h_{R1D}^{j,i_1} (x_{t_2}^{i_{R1}} - \hat{x}_{t_2}^{i_{R1}}) + \sum_{i_{R2}=1}^{n_{R2}} h_{R2D}^{j,i_1} (x_{t_2}^{i_{R2}} - \hat{x}_{t_2}^{i_{R2}}) \right|^2 \quad (\text{Eq. 5-37})$$

假如其中一個 relay 端解碼失敗，接收端處理兩個相異 coded symbol，在星狀圖上的距離，用下列式子表示：

$$\sum_{j=1}^{n_D} \sum_{t_2=L+1}^{2L} \left| \sum_{i_{R1}=1}^{n_{R1}} h_{R1D}^{j,i_1} (x_{t_2}^{i_{R1}} - \hat{x}_{t_2}^{i_{R1}}) \right|^2 \quad \text{or} \quad (\text{Eq. 5-38})$$

$$\sum_{j=1}^{n_D} \sum_{t_2=L+1}^{2L} \left| \sum_{i_{R2}=1}^{n_{R2}} h_{R1D}^{j,i_1} (x_{t_2}^{i_{R2}} - \hat{x}_{t_2}^{i_{R2}}) \right|^2 \quad (\text{Eq. 5-39})$$

同樣根據[9]，利用 source 與 relay 間的 exact 錯誤率，把此系統的錯誤率列於下方：

$$P_e = P\{\text{both relays decoding fail}\} + P\{\text{one relay decoding fail}\} \cdot P\{\text{destination decoding fail}\} + P\{\text{relays decoding successfully}\} \cdot P\{\text{destination decoding fail}\}$$

$$\begin{aligned} &\leq P(\mathbf{X}, \hat{\mathbf{X}}) \cdot P(\mathbf{X}, \hat{\mathbf{X}}) \cdot + \\ &P(\mathbf{X}, \hat{\mathbf{X}}) \cdot (1 - P(\mathbf{X}, \hat{\mathbf{X}})) \cdot \left(\prod_{m=1}^{n_{R1}} \left(1 + \frac{E_s}{4N_0} \beta_m \right) \right)^{-n_D} + \\ &P(\mathbf{X}, \hat{\mathbf{X}}) \cdot (1 - P(\mathbf{X}, \hat{\mathbf{X}})) \cdot \left(\prod_{n=1}^{n_{R2}} \left(1 + \frac{E_s}{4N_0} \lambda_n \right) \right)^{-n_D} + \\ &(1 - P(\mathbf{X}, \hat{\mathbf{X}})) \cdot (1 - P(\mathbf{X}, \hat{\mathbf{X}})) \cdot \left(\prod_{m=1}^{n_{R1}} \left(1 + \frac{E_s}{4N_0} \beta_m \right) \right)^{-n_D} \cdot \left(\prod_{n=1}^{n_{R2}} \left(1 + \frac{E_s}{4N_0} \lambda_n \right) \right)^{-n_D} \\ &\leq \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_{R1}} \cdot \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_{R2}} + \left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_{R2}} \cdot \left(\prod_{m=1}^{n_{R1}} \left(1 + \frac{E_s}{4N_0} \beta_m \right) \right)^{-n_D} + \\ &\left(\prod_{l=1}^{n_S} \left(1 + \frac{E_s}{4N_0} \alpha_l \right) \right)^{-n_{R1}} \cdot \left(\prod_{n=1}^{n_{R2}} \left(1 + \frac{E_s}{4N_0} \lambda_n \right) \right)^{-n_D} + \left(\prod_{m=1}^{n_{R1}} \left(1 + \frac{E_s}{4N_0} \beta_m \right) \right)^{-n_D} \cdot \left(\prod_{n=1}^{n_{R2}} \left(1 + \frac{E_s}{4N_0} \lambda_n \right) \right)^{-n_D} \end{aligned} \quad (\text{Eq. 5-40})$$

由上述的推導可知，假如 $n_{R1}=n_{R2}=n_R$ 、此系統的 diversity 為 $\min\{2n_S n_R, (n_S+n_D)n_R, 2n_R n_D\}$

5.4.2.4 比較前面三節的封包錯誤率：

模擬 $1e7$ 次的實驗次數，每次實驗傳送一個封包，封包的大小為 260bits，調變為 QPSK，分散式籬柵空時碼 係數表示如下，

分散式籬柵空時碼 states	Generator sequence (Source)	Generator sequence (Relay 1)	Generator sequence (Relay 2)
4-states	$g^1=[(00),(20)]$ $g^2=[(00),(10)]$	$g^1=[02]$ $g^2=[01]$	$g^1=[20]$ $g^2=[10]$

- 不同通道環境下，觀察 5.2.4.1 的模擬結果， $(\gamma_{S-D}, \gamma_{S-R}, \gamma_{R-D})$ 分別代表 source-destination、source-relay、relay-destination 的訊雜比。(單位：dB)

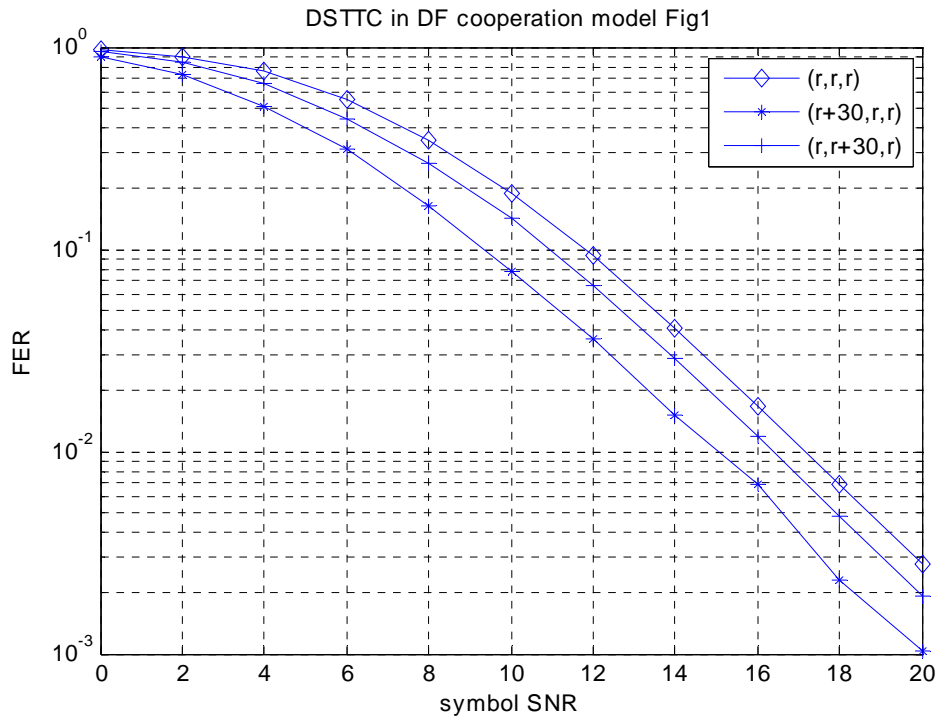


圖 5-11 5.2.1 節增加分散式籬柵空時碼保護後，封包錯誤率表現

- 不同通道環境下，觀察 5.2.4.2 的模擬結果， $(\gamma_{S-D}, \gamma_{S-R}, \gamma_{R-D})$ 分別代表 source-destination、source-relay、relay-destination 的訊雜比。

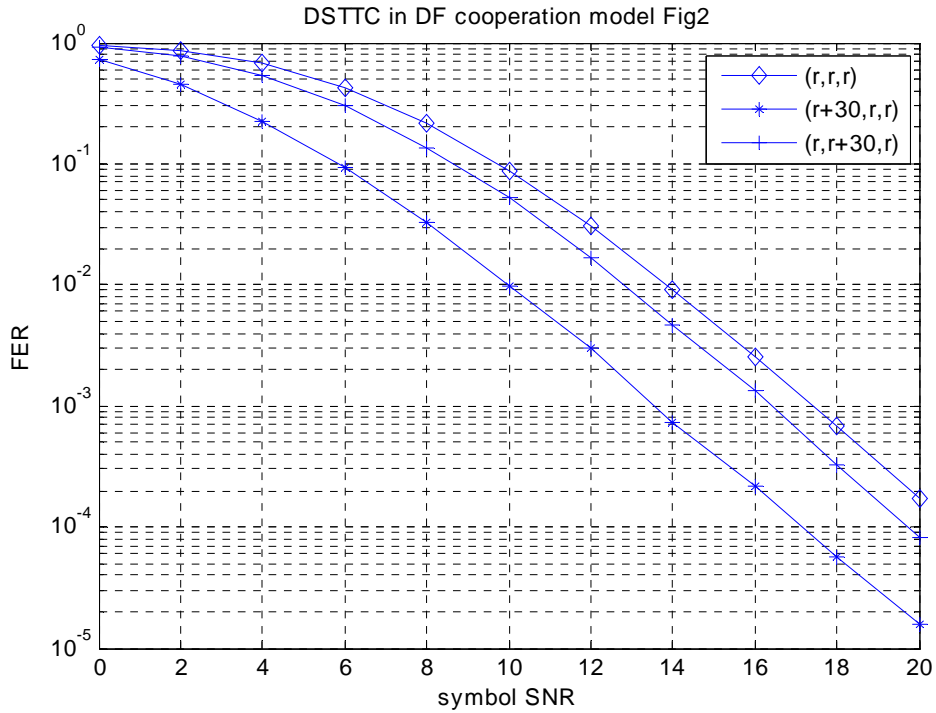


圖 5-12 5.2.2 節增加分散式籬柵空時碼保護後，封包錯誤率表現

- 不同通道環境下，觀察 5.2.4.3 的模擬結果， $(\gamma_{S-D}, \gamma_{S-R}, \gamma_{R-D})$ 分別代表 source-destination、source-relay、relay-destination 的訊雜比。

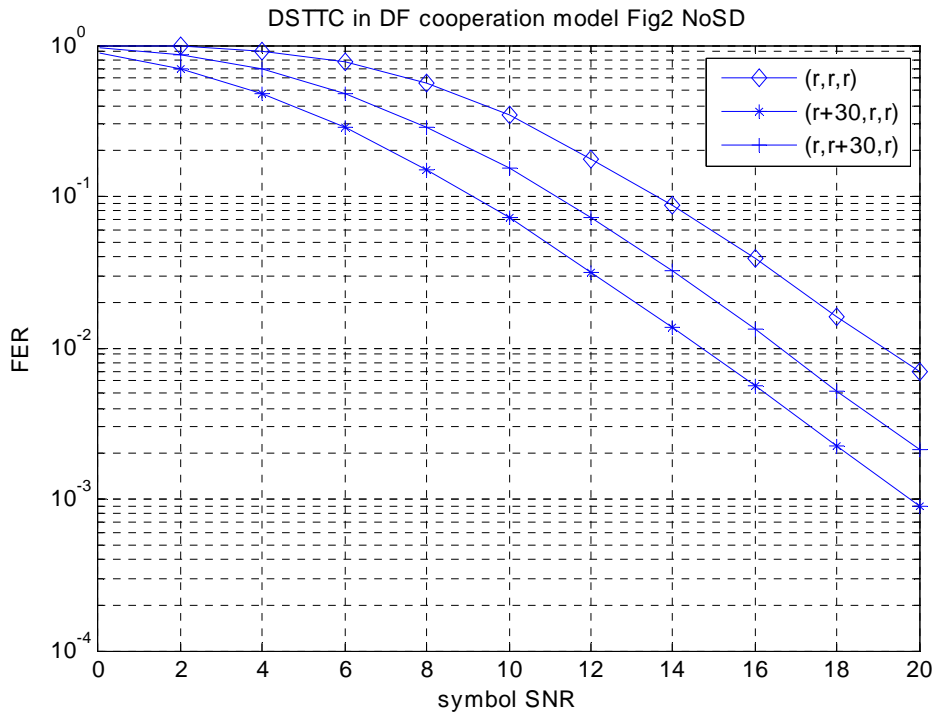


圖 5-13 5.2.3 節增加分散式籬柵空時碼保護後，封包錯誤率表現

- 比較 3 種 system model 的效能。

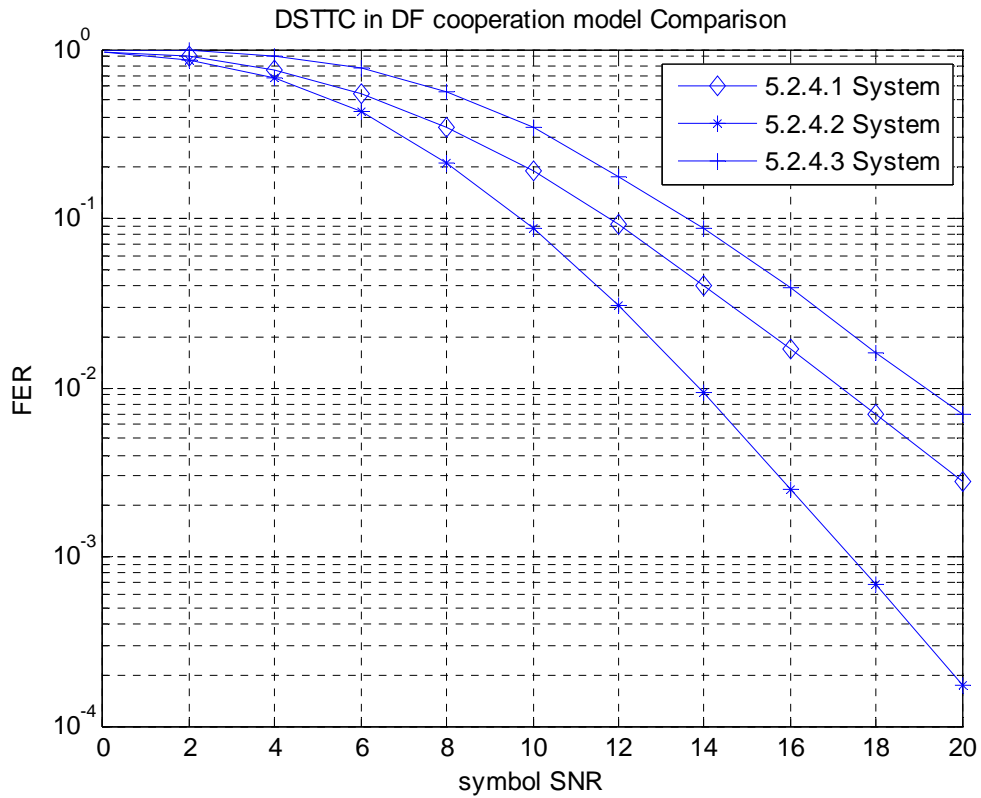


圖 5-14 比較三種傳輸方式的封包錯誤率表現

由上述的模擬結果，可以得知，5.2.4.2 的傳輸方式，較為有效率，且它具有較大的 diversity；此外在通道訊雜比存在差異化時，source-relay 的通道狀態顯得較為重要，由模擬發現，source-relay 得到好的通道係數，比起 relay-destination 得到好的通道係數來的有用，這是因為 error propagation 的關係。

第六章 分散式籬柵空時碼接收機設計

當系統電路問題，導致某一傳送天線所根據的編碼器狀態(state)與其他天線不同；或者另一種情況，DF(Decode and Forward)合作式通訊中，傳送端與中繼端間的通道受嚴重干擾，中繼端無法正確偵測訊號(information bits)，導致中繼端編碼器對於錯誤的資訊作編碼，進而使接收端解碼錯誤。受到干擾的中繼端，其所編碼的資訊會與其他正常中繼端不同，故受干擾中繼端的編碼狀態過程轉換也與其他正常狀態的中繼端不同，造成中繼端群編碼時，發生狀態不一致的問題。接收端收到的訊號，進入傳統籬柵解碼器時，受干擾中繼端訊號影響，接收端在計算路徑距離時會有偏差；甚至在無 AWGN 情況下，傳統籬柵解碼的籬柵下，根本無法找到一條 zero cost 的路徑，容易造成接收端解碼錯誤。本章節提出此類問題的解決辦法—分離式解碼與混合式籬柵解碼。以下的說明，為了方便起見，以合作式通訊中，傳送端與中繼端間通道品質惡劣所造成的編碼器狀態不一致問題為例子說明，系統存在三個中繼端，並限制傳送端、中繼端和接收端僅有單根天線；當某一傳送端與中繼端間的通道受嚴重干擾時，導致中繼端群編碼器狀態不一致的問題，我們稱此中繼端為”受干擾中繼端”。

6.1 系統架構

系統有 n_T 個中繼端，在時間點 t ，中繼端群的天線輸出訊號定義為 $\mathbf{x}_t = [x_t^1 \ x_t^2 \ \dots \ x_t^{n_T}]^T$ ($[\]^T$ 代表轉置矩陣)，這些訊號經過通道 \mathbf{H}_t 的衰減干擾和白色高斯雜訊的影響後，由 n_R 根接收天線接收， n_R 根接收訊號定義為 $\mathbf{r}_t = [r_t^1 \ r_t^2 \ \dots \ r_t^{n_R}]^T$ ，而 \mathbf{r}_t 和 \mathbf{x}_t 的關係可以用下列式子表示：

$$\mathbf{r}_t = \mathbf{H}_t \mathbf{x}_t + \mathbf{n}_t \quad (\text{Eq. 6-1})$$

其中 \mathbf{H}_t 是一個 $n_R \times n_T$ 的通道矩陣，且假設接收端已經知道通道狀況 (\mathbf{H}_t)， h_{ij} 是 \mathbf{H}_t 矩陣的第 i 列、第 j 行的元素，也就是從第 j 根傳送天線到第 i 根接收天線間的通道增益係數，這些元素都是平均值 (mean) 等於零，變異數 (variance) 等於一的 independent circularly symmetric complex Gaussian random variable。 \mathbf{n}_t 是一個 $n_R \times 1$ 的雜訊向量， $\mathbf{n}_t = [n_t^1 \ n_t^2 \ \dots \ n_t^{n_R}]^T$ ，且裡面的元素都是平均值等於零，變異數等於 σ^2 的 independent circularly symmetric complex Gaussian noise。本章節中，討論類慢速衰減 (quasi slow fading) 通道，在一個 frame size 內，通道不會變化，frame size 間的通道係數彼此不影響，視為獨立產生，因此在一個 frame size 的時間內 $\mathbf{H}_t = \mathbf{H} \ (t=1,2,\dots,L \ L=\text{frame size})$ 。當接收天線收到訊號後，收到的訊號 \mathbf{r}_t 會在接收端進行傳送訊號的估測與解碼。籬柵空時碼典型的解法方式採用維特比演算法 (Viterbi)，執行 add、compare、select。其中 branch metric 的計算方式，表示如下：

$$\sum_{j=1}^{n_R} \left| r_t^j - \sum_{i=1}^{n_T} h_{j,i}^t x_t^i \right|^2 \quad (\text{Eq. 6-2})$$

每經過一個 frame size 的時間後，維特比演算法會估計出一組訊號向量，此估計訊號的特性，就是與接收訊號的向量距離最為接近的一組值。維特比演算法屬於 ML 演算法的一種簡化，效果與 ML 相同，但運算速度效能比 ML 佳，是現代通訊技術中常用的演算法。

通訊系統中，其中有一個中繼端，因與傳送間的通道不良，造成中繼端對傳送端資訊解碼錯誤，中繼端對於錯誤的訊號做編碼，再傳給接收端，導致在沒有雜訊的狀況下，接收端即存在解碼錯誤的風險。如圖 6-1，假設 R_3 在 phase 1 時， R_3 與中繼端間的通道受干擾嚴重，因而在 phase 2 時傳送錯誤的訊號給接收端。

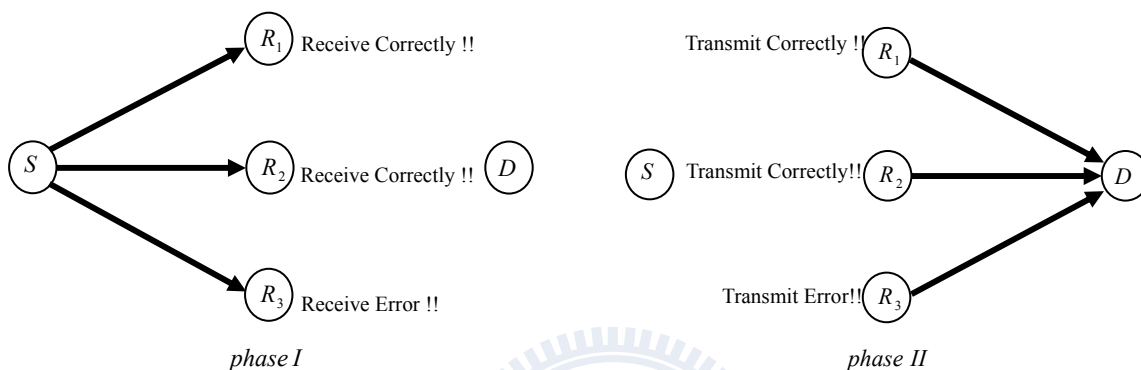


圖 6-1 系統存在惡意中繼端的傳送訊號模式

接收端，使用前篇章節方法，運用維特比(Viterbi) 解碼。因為 R_3 受干擾的關係，導致解碼判斷錯誤，並非選擇 trellis 上，正確的路徑。在此我們試圖尋找些方法，有效解決單一中繼端在 phase 1 受嚴重干擾，在 phase 2 傳錯誤的信號給目的端，造成解碼錯誤的問題。

6.2 傳統接收端解碼方式所面臨的問題

傳統 MIMO 上，每一根接收天線所收到的訊號，為所有傳送天線的傳送訊號值與高斯雜訊累加之和。如下面圖 6-2 所示：

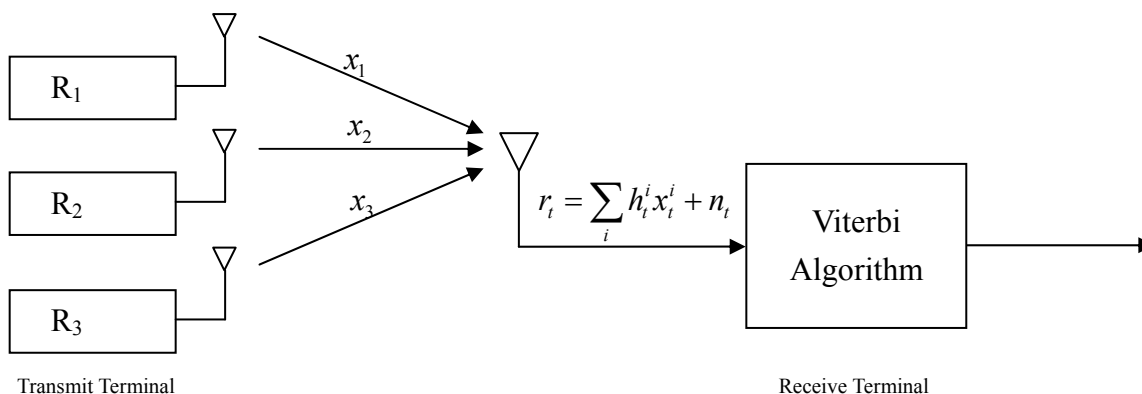


圖 6-2 分散式系統傳收端的區塊圖

維特比解碼中，所使用的 trellis，建造方式，是根據三個中繼端對相同的資訊所建立。假設通道環境中，不存在高斯雜訊，也不存在受干擾中繼端，接收端已知通道係數的情況下，使用維特比演算法解碼，必定會存在一條 zero cost path 的路徑。然而，存在一個受干擾中繼端時，無高斯雜訊的狀況下，因為搜尋不到 zero cost 的路徑，所以也無法百分之百保證是某一個 information bits。觀察例子，參考圖 3-3 之 DSTTC encoder，圖 6-3、圖 6-4 為對應的解碼籬柵。無 AWGN 影響下，傳送端欲傳 all zero 資訊給接收端，正常狀況下，圖 6-3 (或圖 6-4) 的 path1 應包含於接收端的籬柵解碼路徑中。假設 R_3 將一個 bit 解碼錯誤，誤以為傳送訊號是 $[010000\dots 0]$ ，因而對於錯誤的資訊重新編碼，再傳給接收端。單一接收天線，無 AWGN 情況下，觀察兩個例子。第一個例子，當 $\mathbf{H}=[1,1,1]$ 時， $\mathbf{r}_t=[2+j, 3, 3, \dots]$ ，使用傳統籬柵解碼方式，觀察圖 6-3，在 $t=3$ 時，path 1 的路徑有機會被保留下來，故接收端有機會解碼成功。

表 6-1 具延遲特性之籬柵空時碼

籬柵空時碼	Generator sequence
Code A	$g^1=[(002),(020),(200)]$ $g^2=[(001),(010),(100)]$

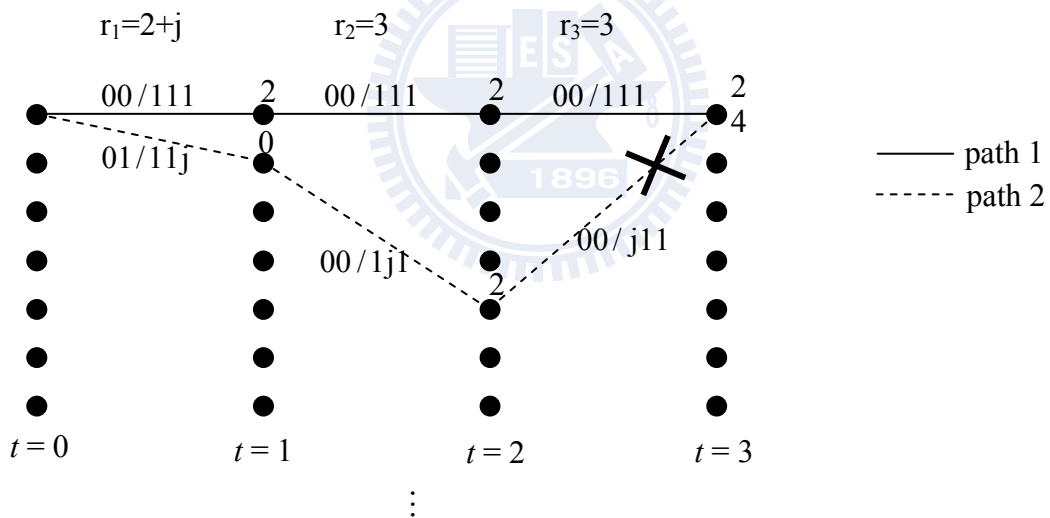


圖 6-3 受干擾中繼端發射錯誤訊號至接收端，造成傳統籬柵解碼錯誤例子一(有機會成功解碼)

第二個例子，當 $\mathbf{H}=[1,1,3]$ 時， $\mathbf{r}_t=[2+3j, 5, 5, \dots]$ ，使用傳統籬柵解碼方式，觀察圖三，在 $t=3$ 時，接收訊號與 path 2 的距離比 path 1 小，故 path 1 的路徑被刪除，接收端沒有機會解碼成功。由此可知，接收端已知通道係數下，仍無法找到一條 zero cost 的路徑，就有解碼錯誤的風險，因為除了編碼以外，通道係數也會影響解碼端，籬柵中 path 與 path 間的距離大小；此外，加上受干擾訊號影響，使得計算接收訊號與 path 間的最小距離而求得真實傳送訊號的方式，更不準確。

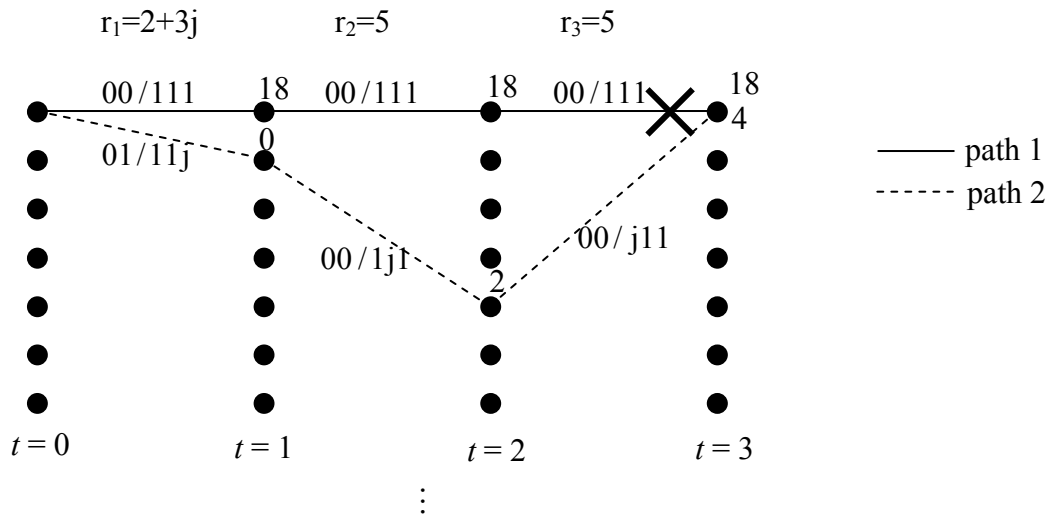


圖 6-4 受干擾中繼端發射錯誤訊號至接收端，造成傳統籬柵解碼錯誤例子二(無解碼成功機會)

使用 Visual C++ 2005 軟體，實驗次數 $1e7$ 次，每次實驗傳遞長度為 20bits 的資訊，接收端解碼時，一個 bit 出錯，計實驗失敗一次。以下為傳統籬柵解碼方式在中繼端群編碼器狀態一致與不一致時，錯誤率比較圖，如圖 6-5。分散式籬柵碼的衍生係數參考表一。中繼端群編碼器狀態一致時，傳統籬柵解碼在錯誤率圖上能夠達到 diversity 為 3 的效能，能充分利用 STC 的特性。當傳統籬柵解碼方式遇到中繼端群編碼器狀態不一致時，失去了作用，儘管中繼端群使用更大的功率傳送訊號，欲降低錯誤率，結果仍是於事無補。

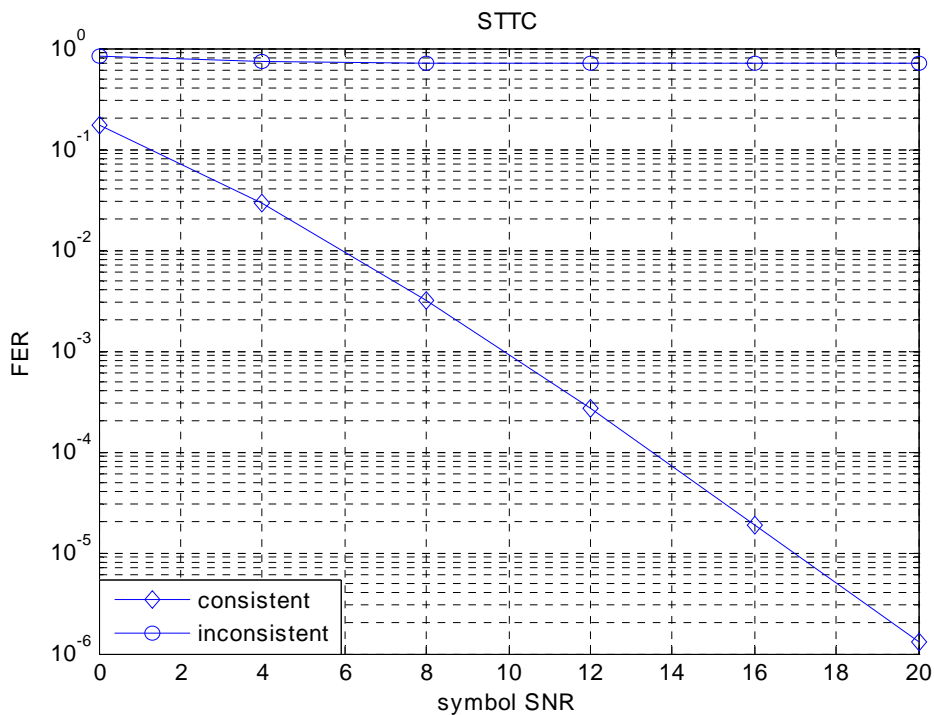


圖 6-5 傳統解碼方法在中繼端群編碼器狀態一致與不一致情況下，錯誤率比較圖。

6.3 利用具延遲特性的籬柵空時碼來解決問題

6.3.1 具延遲特性的籬柵空時碼

有延遲特性的籬柵空時碼，使得傳送端的三根天線，所製造出來的訊號矩陣，具備 full diversity；此類空時編碼，第 i 根天線於 $i-1$ 時間以前，皆不會有訊號(或以一個 dumb 訊號替代，如"1")，constraint length 與 difference 矩陣中 row 的個數相同，如下式子(Eq. 6-2)：

$$\mathbf{B} = \begin{bmatrix} x_1^1 & x_2^1 & x_3^1 & x_4^1 & x_5^1 & \cdots \\ 1 & x_2^2 & x_3^2 & x_4^2 & x_5^2 & \cdots \\ 1 & 1 & x_3^3 & x_4^3 & x_5^3 & \cdots \end{bmatrix} \quad (\text{Eq. 6-3})$$

所以任何兩相異的 code matrix difference，會變成如下

$$\mathbf{B} - \hat{\mathbf{B}} = \begin{bmatrix} x_1^1 - \hat{x}_1^1 & x_2^1 - \hat{x}_2^1 & x_3^1 - \hat{x}_3^1 & x_4^1 - \hat{x}_4^1 & x_5^1 - \hat{x}_5^1 & \cdots \\ 0 & x_2^2 - \hat{x}_2^2 & x_3^2 - \hat{x}_3^2 & x_4^2 - \hat{x}_4^2 & x_5^2 - \hat{x}_5^2 & \cdots \\ 0 & 0 & x_3^3 - \hat{x}_3^3 & x_4^3 - \hat{x}_4^3 & x_5^3 - \hat{x}_5^3 & \cdots \end{bmatrix} \quad (\text{Eq. 6-4})$$

由上述的 difference 矩陣可以看出來，具延遲特性的籬柵空時碼的差異矩陣具備 full diversity。

6.3.2 接收端的接收方式

惡意的中繼端存在下，為了解碼方便，避掉通道的影響，先藉由 LST 的解碼方法，將每根傳送天線上的訊號解出來，再將此經過 LST 解碼方式的訊號，利用維特比演算法解出傳送端的資訊(informaiton bit)。以下圖 6-5 為解碼端的區塊圖：

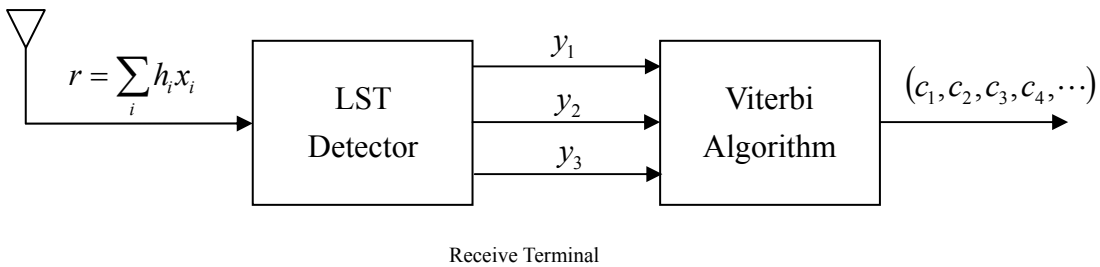


圖 6-6 使用具延遲特性的籬柵空時碼來解決問題時，所使用的接收端架構

圖 6-6 中的 y_1 、 y_2 、 y_3 為接收端估計三個中繼端所傳訊號的結果，並不一定與原來三個中繼端所傳的訊號相同， $(x_1, x_2, x_3) \neq (y_1, y_2, y_3)$ ；因此，高斯雜訊對訊號的影響結果，在此已經表現出來。維特比演算法的目的，是為了判斷何者為傳送端原本傳送的訊號，即便在有受干擾中繼端搗亂或雜訊影響的情況下，也能正確判斷。

$$\text{branch metric} = \sum_{i=1}^{n_T} |y_i - \hat{x}_i|^2 \quad (n_T \text{ 為中繼端的個數}) \quad (\text{Eq. 6-5})$$

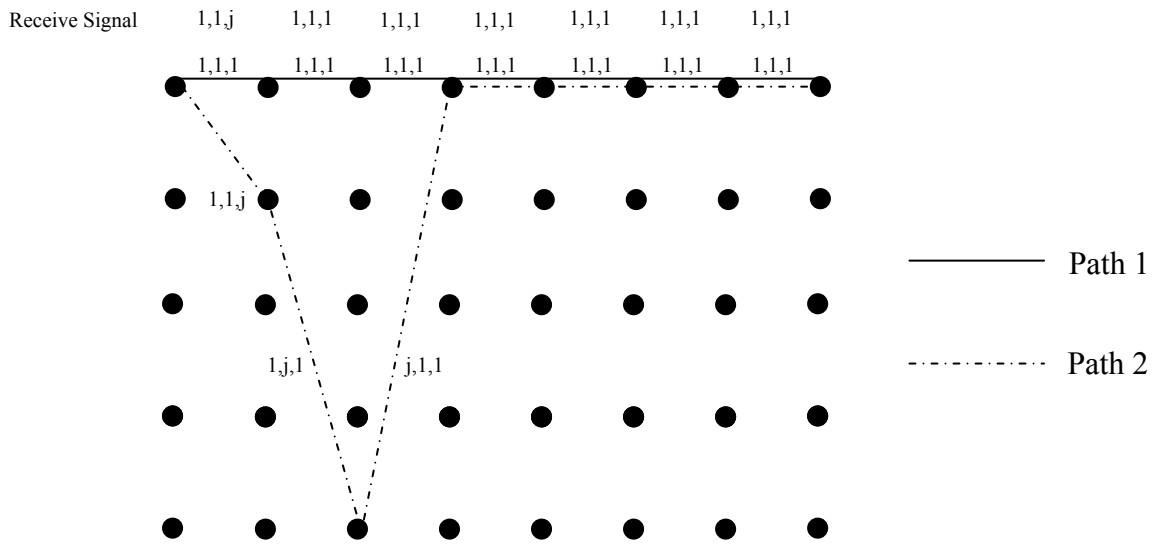


圖 6-7 經過 LST detector 後，計算中繼端的方法。

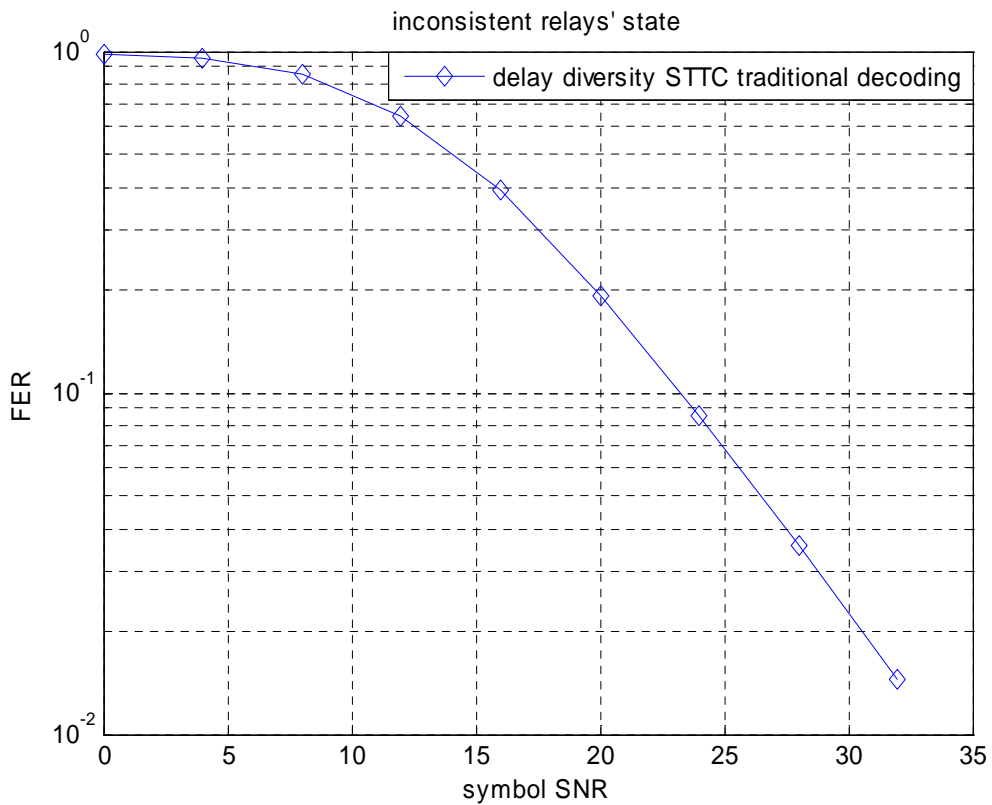


圖 6-8 傳送端使用 delay diversity STTC，接收端使用圖 6-6 方式來解決中繼端群狀態不一致問題，其錯誤率表現圖。

另一點要注意的是，此時維特比解碼的方式，與前述幾章節所使用的方式，有稍微

不同。由於進入 Viterbi Block 的訊號，為經過 LST Detector 後的訊號，所以計算維特比中的 branch metric 時，等同計算三個中繼端與估計訊號間的 symbol-by-symbol 距離平方，如 Eq. 6-5，配合圖 6-6 所示：

接收訊號與 path 1 上的訊號距離平方為： $|j-1|^2 = 2$

接收訊號與 path 2 上的訊號距離平方為： $|1-j|^2 + |1-j|^2 = 4$

當 branch metric 的計算方法定義完畢之後，就可以仿照傳統的維特比演算法，執行 Add-Compare-Select，選取與接收訊號最短的路徑。

根據實驗的結果發現，具備延遲特性的籬柵空時碼，必定能糾正一個中繼端的錯誤，由於圖 6-6 的解碼方式也不屬於 STC ML 的解法，所以錯誤率表現也不如中繼端群編碼器狀態一致時，使用傳統籬柵解碼方法的結果(錯誤率 diversity 為 3)。模擬方法同 6.2 節方式，圖 6-8 顯示傳送端使用 delay diversity STTC(衍生係數參考表一)，接收端使用圖 6-6 方式來解決中繼端群狀態不一致問題，其錯誤率表現。

6.4 分離式 Viterbi 解法

圖 6-9 為分離式解碼器區塊圖，LST detector 代表在時間 $t (t=1 \sim L)$ 、高斯雜訊(AWGN)模型下，偵測各個傳送天線符碼的估測器，計算方法可以由接收訊號與估測訊號在星狀圖的距離來計算，如下式：

$$distance = \|\mathbf{r}_t - \mathbf{H}\mathbf{y}_t\|^2 \quad (\text{Eq. 6-6})$$

其中， \mathbf{y}_t 以所有可能的訊號組合代入(全域搜尋法)，與 \mathbf{r}_t 的最小距離，即為最佳解。由於接收端訊號經過 LST detector 後，能夠區分出三個中繼端的傳送資料($\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3$)；假設三個中繼端各自建造自己的籬柵，稱為 trellis 1、trellis 2、trellis 3，然後分別將 \mathbf{y}_1 、 \mathbf{y}_2 、 \mathbf{y}_3 ，輸入 trellis 1、trellis 2、trellis 3 作維特比解碼，即可得到三個中繼端的編碼器輸入資訊；在無雜訊的干擾之下，能將正確訊號與錯誤訊號都偵測出來。當接收端只在乎傳送端資訊時(不在乎錯誤資訊)，能再根據三個中繼端的解出資訊，分別做檢查。當三個中繼端的資訊的其中之二，與傳送端的資訊相同，即算傳送成功。限制需要兩個資訊通過檢查的原因，是由於受干擾中繼端所傳的資訊也為合理資訊，也能通過檢查。

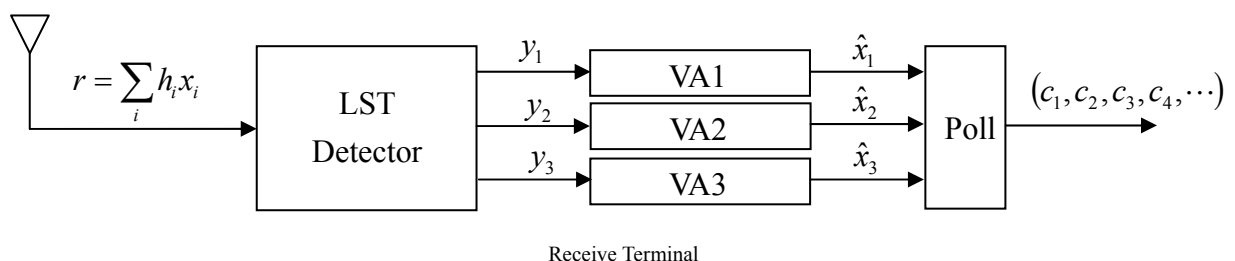


圖 6-9 分離式維特比解法的架構圖

從另一個觀點來看，此種解碼方式，三個中繼端的訊號，視為彼此不相關連的訊號，經由各自的編碼器，形成三個相異天線的發射訊號，接收端的訊號輸入 LST Detector 之後，

將每一個時間點的發射訊號 (y_1, y_2, y_3) ，估計出來；但此時的 (y_1, y_2, y_3) 不享有 DSTTC 的保護， (y_1, y_2, y_3) 估計的誤差，是三個中繼端到接收端的通道響應、白色高斯雜訊與符碼間的干擾等所造成；因為 detect 時，我們會根據接收訊號與估計訊號在星狀圖上的距離，作適當的量化(quantization)，取最接近原始訊號的組合，因此這部分不僅有白色高斯雜訊造成的誤差，也可能會有量化所造成的誤差。此外， (y_1, y_2, y_3) 各自解碼，而 y_1 經過 y_2 、 y_3 間的相互干擾後， y_2 、 y_3 依此類推，才進行解碼，這部分已與原來空時碼所使用 ML 方法不同，故錯誤率表現，已不可用 Eq. 2-20 或 Eq. 2-21 來判端，但可以猜測的是，錯誤率表現較差，diversity 特性降低。

以上的想法，與 VBLAST 解碼器類同，只是在前後增加了類似迴旋碼的編碼器與解碼器。面對傳送端使用籬柵空時碼，這樣的解碼方法，當傳送端與中繼端間的通道受干擾嚴重時，也能提供一種解決方式，近似 multiple access 功能。依據[1]，LST 強調的是多工(multiplexing)的特性，LST 在 diversity 的表現上，必須靠 receiver antenna 來加強，由於這裡假設傳送端、中繼端和接收端僅有單根天線，所以這裡的錯誤率表現，儘管使用了籬柵空時碼的概念，但不具有 full diversity 特性。

6.5 混合式(Hybrid State)籬柵解碼

為了解決中繼端群編碼器狀態不一致問題，並且對於受干擾中繼端輸入訊號也能做出正確的判斷，因此提出混合式籬柵解碼。因為受干擾中繼端所編的資訊(information bits)與正常中繼端群所編的資訊不同，所以在建立 DSTTC 的籬柵時，同時也須考慮“受干擾中繼端”的輸入資訊(information bits)；我們在建立籬柵時，是以整個分散式系統觀點切入，將所有中繼端群看成一個大編碼器，使用籬柵來紀錄此大編碼器的輸出輸入關係。

6.5.1 混合式籬柵解碼複雜度討論

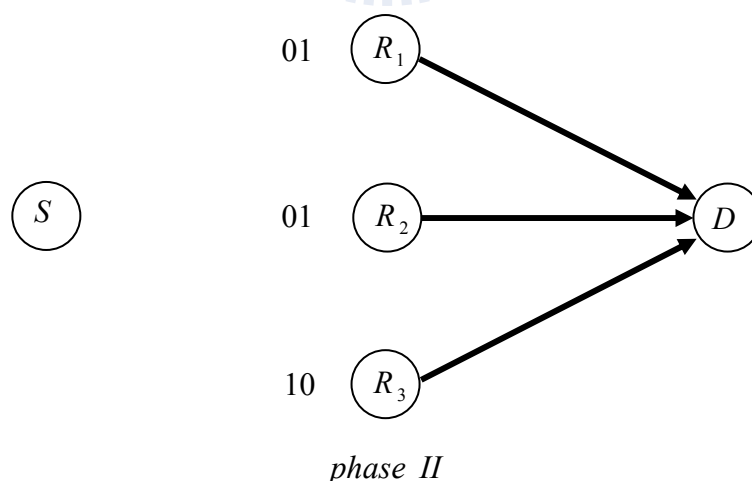


圖 6-10 惡意中繼端對於不同的 information bits 編碼的示意圖

當系統存在一個受干擾中繼端時，因為受干擾中繼端所認定之傳送端資訊，相異於正常中繼端，所以受干擾中繼端會傳送與其他正常中繼端不同的資訊(information bits)。以 QPSK 籬柵空時碼為例，當系統存在一個受干擾中繼端(R_3)時，受干擾中繼端會傳送

與其他中繼端(R_1 、 R_2)不同的 information bits，參考圖 6-10。在每一個時間點，正確的資訊(information bits)與受干擾中繼端的錯誤輸入資訊組合起來，總共有 16 種組合，這也造成了籬柵上的 state diagram，由每一個狀態所延伸出來的 branch 共 16 條，不同於原來僅需的 4 條，增加的原因，是為了辨認受干擾中繼端所發出的資訊；此外，假如考慮每個中繼端都有機會成為受干擾中繼端時，籬柵圖上由每個狀態延伸出來的 branch 會更多。以三個中繼端使用 QPSK 籬柵空時碼為例，籬柵圖中，由每個狀態延伸出來的 branch 會變為 48 條，是未紀錄何者中繼端為受干擾中繼端時的三倍。

另外一個因中繼端群編碼器狀態不一致，使得混合式籬柵碼與傳統籬柵碼相異的地方為 trellis diagram 必須記下整體分散式系統的狀態。傳統無受干擾中繼端的 Feed-Forward 籬柵空時碼，因為全部中繼端皆對相同的資訊(information bits)作編碼，所有中繼端的狀態變化皆相同，解碼時，就能利用這個特性，作維特比解碼；然而，存在受干擾中繼端的狀況下，受干擾中繼端裡暫存器的狀態變化過程，可能與正常中繼端裡暫存器的狀態變化過程不同。為了解決這種問題，建立籬柵時，我們必須將受干擾與正常中繼端的狀態組合起來。吾人將編碼器狀態不一致時，稱此系統的大編碼器於”延伸狀態”。由於受干擾中繼端的狀態變化，並不屬於原本傳送端欲傳資訊應產生的狀態變化，這些因組合而多出來的”虛擬”狀態，加上原有正常的狀態(所有中繼端編碼狀態一致)，組合起來的編碼系統，合稱為混合式籬柵解碼(Hybrid trellis decoding)。舉一個 QPSK DSTTC 例子，當系統存在三個中繼端時，示意圖如圖 6-11：

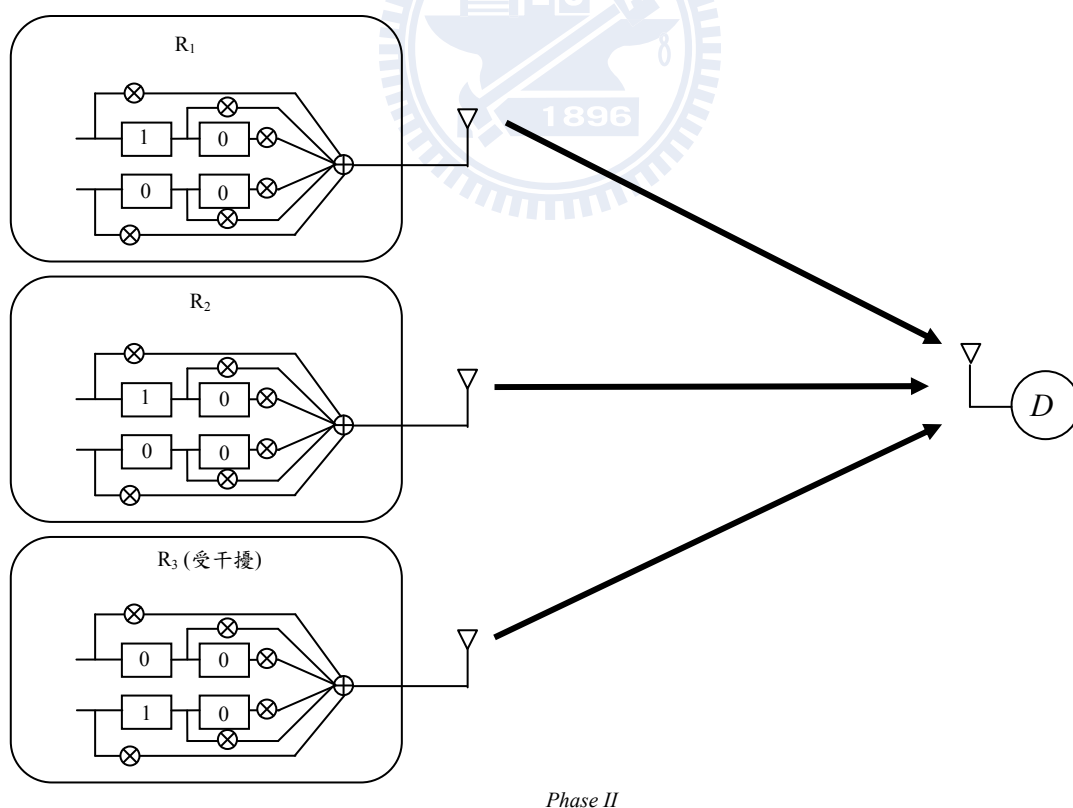


圖 6-11 三個中繼端在不同的 state 對應到新的 state 示意圖

以具有 16 種狀態的 QPSK 籬柵空時碼為例，假設三個中繼端皆正常的話，三個中繼端依同一個資訊作編碼，對於 Feed Forward 籬柵空時碼，在每個時間點皆會在相同的狀態，傳統籬柵解碼有能力解出此類編碼方式的訊號，是因為接收端的籬柵僅紀錄三個編碼器在相同狀態時的籬柵關係。然而，圖 6-11 中， R_1 、 R_2 為正常中繼端，傳相同的資訊時，兩個中繼端都會在相同的狀態；而 R_3 為受干擾中繼端，故與 R_1 、 R_2 在不同的狀態。為了紀錄整個分散式通訊系統的狀況，我們必須將正確與錯誤的狀態結合在一起，成為新的狀態，對應(mapping)方式可以參考如下 $(R_1 \text{ state}, R_2 \text{ state}, R_3 \text{ state}) = (2, 2, 1) \rightarrow 221_{16} = 545_{10}$ ；也就是說，假如 R_1 、 R_2 原本在 $\text{state}=2$ 、 R_3 原本在 $\text{state}=1$ ，將這種情況對應到混合式籬柵碼時，它們所代表系統的狀態為 $\text{state}=545$ 。因此，讓正常的狀態與錯誤的狀態作組合，產生新的狀態，接收端所使用暫存器的數目，就必須要大於各個中繼端所使用的暫存器數目。相較於分離式解碼方法接收端只需用 16 種狀態的暫存器解碼，混合籬柵解碼則必須至少使用 $16^2 \times 3$ 種狀態的暫存器解碼。這樣的設計方法，假如在無高斯雜訊與接收端已知通道係數的狀況下，接收端必能找到一條 zero cost 的路徑，恰巧符合三個中繼端真實所傳送的訊號。然而，真實的通訊環境中，存在高斯雜訊，接收端即有估測訊號的困擾，在此不仿效前面章節使用 LST detector，直接將接收訊號輸入擁有混合籬柵的 Viterbi decoder，執行 joint 解碼，參考圖 6-12。沒有 LST detector，直接進入 Viterbi decoder 的好處，是除去了量化誤差所造成的影響，執行籬柵空時碼真正的 ML 解碼，使得解碼效能較擁有 LST detector 時的好。第三章需要 LST detector 的原因，是由於我們必須得到三個傳送天線的值。最後提醒者，混合籬柵碼所解出的資訊，是三個中繼端的輸入資訊，因此混合籬柵碼不僅能夠解出正確資訊，也能偵測錯誤資訊。

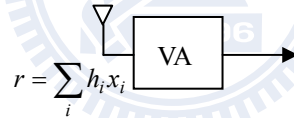


圖 6-12 混合式籬柵解碼(ML)示意圖

6.5.2 混合式籬柵解碼分級程度討論

所有的籬柵空時碼，必存在 all zeros information bits，其對應的 coded symbol 為 all value one (Eq. 6-7)。

$$\text{Information bits}=(0,0,\dots,0) \rightarrow C_1 = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \end{bmatrix}_{3 \times L} \quad (\text{Eq. 6-7})$$

假使在無受干擾中繼端的狀況下，不管具不具有 full diversity 的 DSTTC，它的 coded symbol 矩陣(Eq. 6-8)表示如下：

$$C_2 = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_L^1 \\ x_1^2 & x_2^2 & \dots & x_L^2 \\ x_1^3 & x_2^3 & \dots & x_L^3 \end{bmatrix}_{3 \times L} \quad (\text{Eq. 6-8})$$

利用 Eq. 2-20 的準則，比較兩個 coded symbol 矩陣。系統欲傳 all zeros information bits，其所對應的 coded symbol 矩陣為 C_1 ，而 R_3 因與傳送端間的通道狀況不良，而對傳送端的資訊解碼錯誤，錯誤資訊經由重新編碼過後，假使傳一組錯誤訊號為 C_2 ，所以將 R_1 與 R_2 的 coded symbol 從 C_1 取出， R_3 的 coded symbol 從 C_2 取出，形成 Eq. 6-9 的 coded symbol 矩陣：

$$C_1^{Error} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ x_1^3 & x_2^3 & \cdots & x_L^3 \end{bmatrix}_{3 \times L} \quad (\text{Eq. 6-9})$$

另外，系統欲傳一組非 all zeros information bits 對應的 coded symbol 為 C_2 ，而 R_2 也因與傳送端間的通道狀況不良，而對傳送端的資訊解碼錯誤，錯誤資訊經由重新編碼過後，假使傳一組錯誤訊號為 C_1 ，所以將 R_1 與 R_3 的 coded symbol 從 C_2 取出， R_2 的 coded symbol 從 C_1 取出，形成以下的 coded symbol 矩陣，Eq. 6-10：

$$C_2^{Error} = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_L^1 \\ 1 & 1 & \cdots & 1 \\ x_1^3 & x_2^3 & \cdots & x_L^3 \end{bmatrix}_{3 \times L} \quad (\text{Eq. 6-10})$$

所以這兩個訊號所形成的 difference coded symbol 矩陣 Eq. 6-11 如下

$$C_2^{Error} - C_1^{Error} = \begin{bmatrix} x_1^1 - 1 & x_2^1 - 1 & \cdots & x_L^1 - 1 \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \end{bmatrix}_{3 \times L} \quad (\text{Eq. 6-11})$$

由上面的矩陣 Eq. 6-11 可以了解，當傳統籬柵解碼具有 full diversity，假使系統存在受干擾中繼端，接收端將傳統籬柵解碼結合虛擬狀態成為混合籬柵解碼，此籬柵空時解碼在通訊系統中，存在錯誤率分集程度為 1 的 $P(X_i, X_j)$ 。對於一個空時碼，它的錯誤率由式子 Eq. 6-12 表示：

$$Error\ Probability = \sum_{j \in U, j \neq i} P(X_i, X_j) \quad (\text{Eq. 6-12})$$

U 代表混合式籬柵中所有路徑的 coded symbol 矩陣集合。式子 Eq. 6-12 中，某些 $P(X_i, X_j)$ 項的 diversity 為 1(如上述例子)，某些 $P(X_i, X_j)$ 項的 diversity 為 2 或 3，這就造成使用混合籬柵解碼時，錯誤率曲線不具有 full diversity 特性，而且在訊雜比很高的情況下，diversity 為 1 的 $P(X_i, X_j)$ 項次，會支配錯誤率曲線的斜率，造成錯誤率曲線的 diversity 很接近 1。總結上述的說明，在中繼端有可能傳送錯誤訊號的狀況下，混和籬柵解碼不存在最大分集程度(full diversity)的傳輸辦法。

使用 Visual C++ 2005，使用混合籬柵解碼方式時，每次實驗輸入 frame size 長度為 20bits 的 information bits，當解碼端偵測到一個 bit 錯誤時，此次實驗即算失敗，累積 100 次的失敗，模擬隨即停止，記錄錯誤率；以累積 100 次的失敗即停止的方式，是避免模

擬時間花費太久。圖 6-13 為在中繼端群編碼器狀態不一致時，混合籬柵碼與傳統籬柵空時碼解碼的錯誤率比較圖；其中分散式籬柵碼的衍生係數可參考表一；驗證存在受干擾中繼端時，混合式籬柵碼能夠正確解碼，用以驗證我們所遇到的問題，其錯誤率表現的 diversity 為 1。然而它的錯誤率不如使用傳統籬柵解碼於中繼端群編碼器狀態一致時的表現。

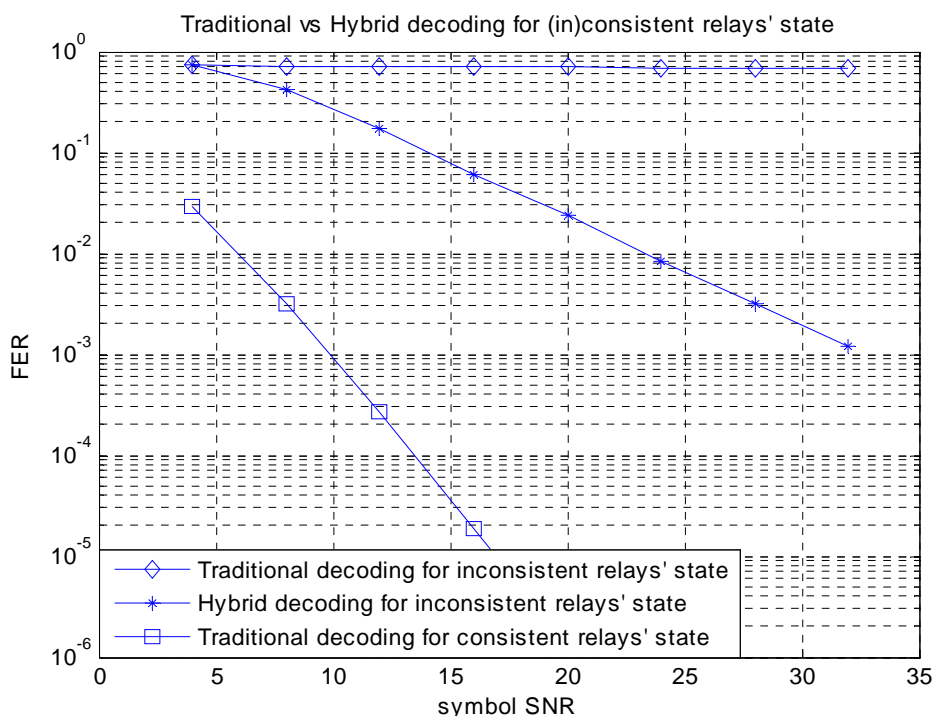


圖 6-13 應用混合式籬柵碼解決中繼端群編碼器狀態不一致問題

6.5.3 混合式籬柵解碼編碼增益討論

本文第二章提及空時碼在慢速衰減通道下的編碼增益，在傳送天線少時，採用 determinant 準則；傳送天線數多時，採用 trace 準則。根據以上的法則，檢視混和籬柵碼所產生的 coded symbol 矩陣，經過矩陣運算之後得到的編碼增益。在 B. 曾經提到，混和籬柵碼分集程度為 1，由矩陣(Eq. 6-11)可知，將 $\gamma=1$ 、 $n_R=1$ 帶入式子(Eq. 2-20)或式子(Eq. 2-21)，最小的 determinant 與 trace 等同於某個中繼端自身所使用錯誤更正碼的編碼增益；在中繼端各自擁有一根傳送天線的情況下，中繼端各自的編碼增益(determinant 與 trace)即為中繼端編碼器各自輸出 coded symbol 間的歐式距離。

此外，根據前面說明，每個中繼端的編碼方式，皆有屬於自己的編碼增益；具有不同編碼增益的中繼端組合在一起，對於整體混合式籬柵碼的表現也有影響。以有序對來表示三個中繼端的編碼增益為(R_1 coding gain, R_2 coding gain, R_3 coding gain)，在混合式籬柵碼中，(4,2,2)的編碼方式會比(2,2,2)的編碼方式表現的好。

圖 6-14 為在中繼端群編碼器狀態不一致時，混合籬柵碼的編碼增益比較，可以發現混合籬柵解碼編碼增益，假如增加某一天線的編碼增益效果，對於混合籬柵解碼編碼

增益也會有所幫助，圖中 Code A 的編碼在 R_1 上的編碼增益小於 Code B 在 R_1 上的編碼增益，所以 Code B 在混合式籬柵解碼方式中，錯誤率表現較 Code A 好。

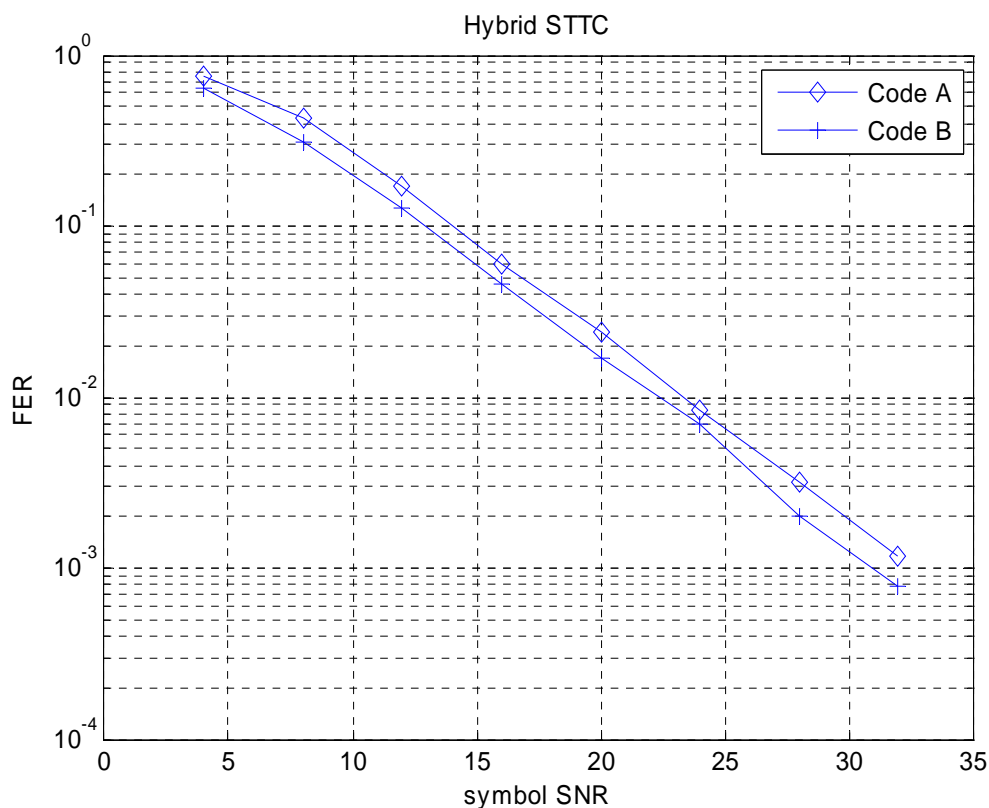


圖 6-14 混合籬柵碼的編碼增益比較

籬柵空時碼	Generator sequence	R_1 min_tr	R_2 min_tr	R_3 min_tr
Code A	$g^1=[(002),(020),(200)]$ $g^2=[(001),(010),(100)]$	2	2	2
Code B	$g^1=[(002),(012),(231)]$ $g^2=[(200),(120),(233)]$	4	2	2

6.5.4 混合式籬柵解碼的多工與分集討論

根據多天線系統的分析，多工與分集的特性是相互消長[10]。在某種程度上，混合式籬柵碼是為了追蹤每個中繼端的輸入信號而產生的演算法；換句話說，在中繼端群編碼器狀態一致時，傳統籬柵空時碼解碼方式，僅追蹤 2 bits 的資訊；然而，在中繼端群編碼器狀態不一致時，混合籬柵空時碼是用來追蹤所有編碼器的輸入資訊，總共為 6 bits，因此混合式籬柵空時碼錯誤率的 diversity 較傳統籬柵空時碼低。

6.5.5 以集合方式討論混合式籬柵解碼—多數決增益 (Voting gain)

混合籬柵碼能夠追蹤每個中繼端的輸入資訊，為了提高正確率，我們將解出的三個中繼端資訊，分別做檢查，當其中兩個中繼端資訊與傳送端的資訊相符時，就算接收端

解碼正確，即訊號傳輸成功，我們稱此解碼方法為”多數決混合式籬柵碼”。需要兩個中繼端資訊通過檢查的原因同第三部分的說明。此方法只求傳送端正確資料無誤抵達接收端，不求追蹤受干擾中繼端的錯誤資訊。當混合籬柵碼能夠偵測中繼端編碼器狀態不一致問題時，混合籬柵碼籬柵中的許多條路徑，會被正確量化成傳統 STTC 上的某一條路徑，示意圖參考圖 6-15。

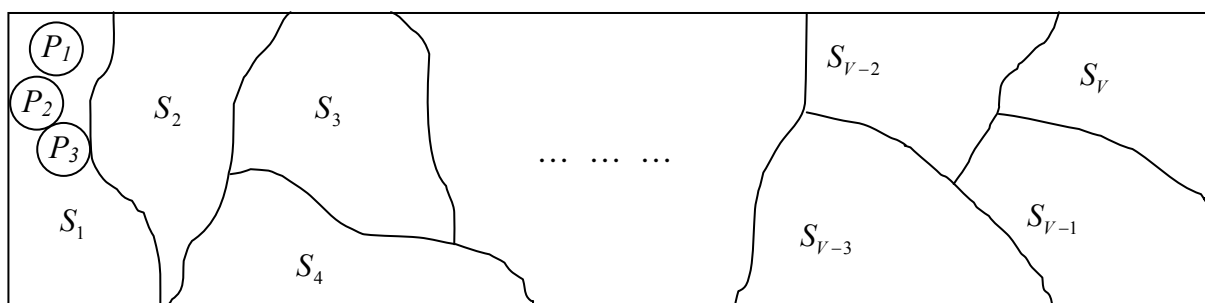


圖 6-15 混合式籬柵解碼與傳統籬柵空時碼解碼的關係圖

S_1 代表傳統 STTC 籬柵的路徑， V 代表傳統籬柵中所有可能路徑個數，也代表 coded symbol 矩陣集合的個數。 P_1 代表使用混合籬柵碼時，籬柵關係圖的其中一條路徑。假使接收端作檢查，使用多數決混合籬柵解碼，混合籬柵中的路徑全部量化在傳統 STTC 的籬柵路徑中。在量化正確的情況下， $S_1 = \{P_1, P_2, P_3, \dots\}$ 代表 P_1, P_2, P_3, \dots 被量化為 S_1 。當 P_1 因雜訊的影響被誤會成 P_2 ，接收端無多數決功能時，在前篇混合籬柵解碼中，為了偵測所有中繼端的輸入訊號，此狀況會被判定為為"錯誤"；然而，在具有多數決功能時，接收端僅需判定傳送端訊號，不需判斷所有中繼端的輸入訊號，具備多數決混合籬柵解碼會被判定此狀況為正確(因為正確量化)， P_1 與 P_2 皆被量化為 S_1 。所以混合式籬柵空時碼加入多數決檢查時，對於錯誤的定義較不嚴苛，理所當然，錯誤率表現會比需偵測所有中繼端輸入資訊時的佳；吾人稱此錯誤率的提升為多數決增益。多數決增益，讓錯誤率表現更為優越。判斷一個空時碼的封包錯誤率由 Eq. 6-12 表示；Eq. 6-12 代表不具備多數決混合籬柵解碼的錯誤率計算方法。錯誤率的分集程度，在訊雜比很高的情況下，通常由 diversity 最小的 $P(X_i, X_j)$ 決定，如同在 6.4.2 章節的證明一般。當使用多數決檢查時，雖然由矩陣(Eq. 6-9)誤會成矩陣(Eq. 6-7)的事件機率，不存在於式子(Eq. 6-12)中，但矩陣(Eq. 6-9)誤會成矩陣(Eq. 6-10)的事件機率依然存在，由矩陣(Eq. 6-11)可以得知，矩陣(Eq. 6-9)誤會成矩陣(Eq. 6-10)的錯誤率 diversity 為 1，所以在錯誤率表現上，具備多數決檢查的混合籬柵碼仍然不具有 full diversity 的特性，錯誤率的 diversity 會接近 1。此外，具備多數決混合籬柵解碼的錯誤率 diversity 可能比不具備多數決時來的高，這是因為某些 diversity=1 的 $P(X_i, X_j)$ 不存在於式子(Eq. 6-12)中。

為了降低錯誤率，我們使用具備多數決混合籬柵解碼方式，只要兩個中繼端的資訊與傳送端資訊相符，即算解碼成功，如此一來能容忍更多的錯誤；如圖 6-16，相較於無具備多數決特性時，具備多數決的解碼方式，能使錯誤率曲線在圖中下移一些，並且在高訊雜比時，錯誤率的 diversity 接近 1。

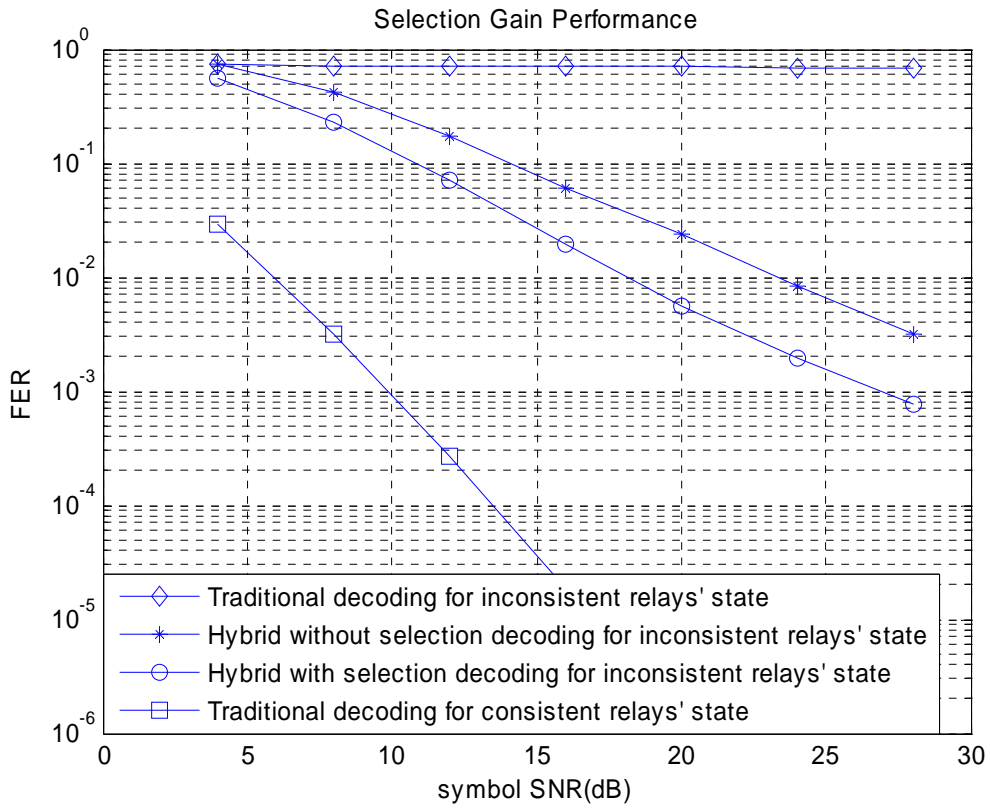


圖 6-16 選擇增益比較圖

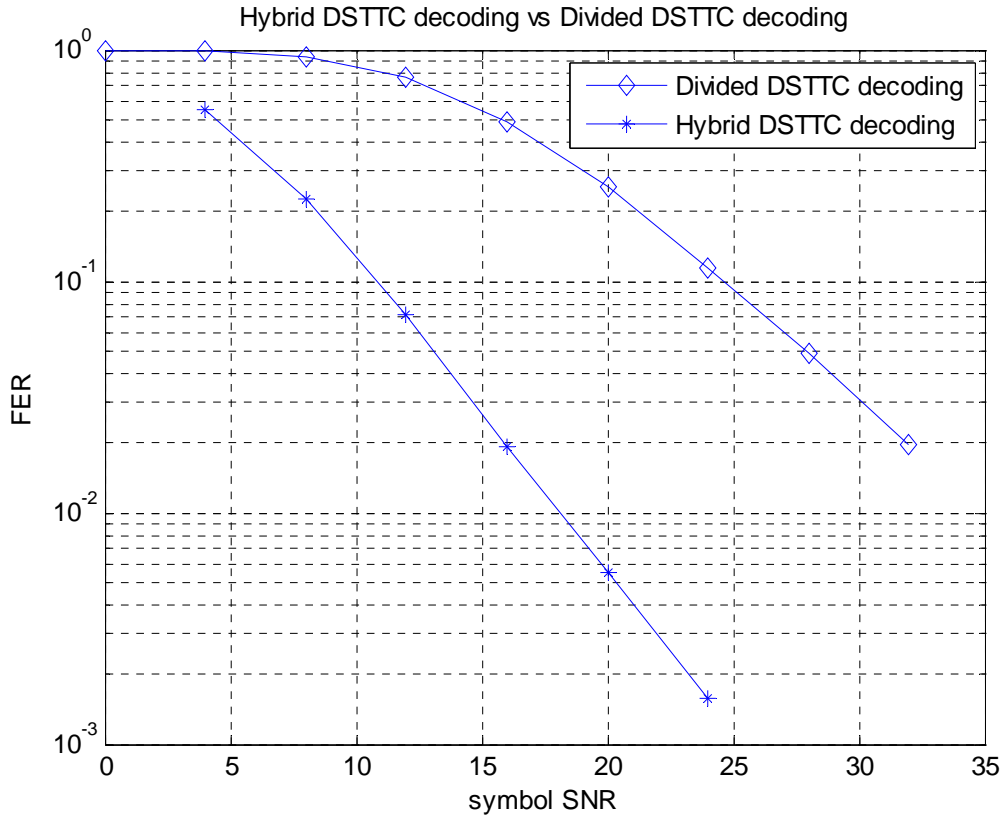


圖 6-17 具備多數決混合籬柵碼與具備多數決分離式籬柵碼的效能比較

圖 6-17 為在中繼端群編碼器狀態不一致時，多數決混合籬柵碼與分離式籬柵碼的效能比較；明顯地，多數決混合式籬柵碼的效能表現優於分離式籬柵碼，效能較好的原因，由於 LST 多了量化誤差與符碼間的干擾，經過 LST 之後才使用維特比演算法，這一部分犧牲了解碼正確率，故造成如此大的誤差。分離式解碼方法並非 ML 解法，故沒有獲得空時碼分集特性的好處。

6.5.6 多數決混合籬柵解碼使用中繼端群編碼器狀態一致下的探討

雖然具備多數決混合籬柵解碼是用來面對中繼端群編碼器狀態不一致的解決辦法。然而，吾人也試著將混合式籬柵解碼方法使用中繼端群編碼器狀態一致時，觀察其錯誤率表現。當中繼端群狀態一致時，代表三個中繼端傳遞相同的資訊。第 k 個中繼端傳遞 \mathbf{X} 資訊時，在一個 frame size 內，其輸出天線訊號為 $\mathbf{X}^k = [x_1^k \ x_2^k \ \dots \ x_L^k]$ 。矩陣 Eq. 6-13 與矩陣 Eq. 6-14 分別代表三跟天線皆傳遞 \mathbf{X} 與 \mathbf{Y} 資訊的 coded symbol 矩陣，其中 \mathbf{X} 與 \mathbf{Y} 為相異資訊，

$$\mathbf{X} \rightarrow \begin{bmatrix} \mathbf{X}^1 \\ \mathbf{X}^2 \\ \mathbf{X}^3 \end{bmatrix} \quad (\text{Eq. 6-13}) \quad \mathbf{Y} \rightarrow \begin{bmatrix} \mathbf{Y}^1 \\ \mathbf{Y}^2 \\ \mathbf{Y}^3 \end{bmatrix} \quad (\text{Eq. 6-14})$$

$$\mathbf{X} \rightarrow \begin{bmatrix} \mathbf{X}^1 \\ \mathbf{X}^2 \\ \mathbf{Y}^3 \end{bmatrix} \quad (\text{Eq. 6-15}) \quad \mathbf{X} \rightarrow \begin{bmatrix} \mathbf{X}^1 \\ \mathbf{Y}^2 \\ \mathbf{X}^3 \end{bmatrix} \quad (\text{Eq. 6-16}) \quad \mathbf{X} \rightarrow \begin{bmatrix} \mathbf{Y}^1 \\ \mathbf{X}^2 \\ \mathbf{X}^3 \end{bmatrix} \quad (\text{Eq. 6-17})$$

$$\mathbf{X} \rightarrow \begin{bmatrix} \mathbf{X}^1 \\ \mathbf{Y}^2 \\ \mathbf{Y}^3 \end{bmatrix} \quad (\text{Eq. 6-18}) \quad \mathbf{X} \rightarrow \begin{bmatrix} \mathbf{Y}^1 \\ \mathbf{X}^2 \\ \mathbf{Y}^3 \end{bmatrix} \quad (\text{Eq. 6-19}) \quad \mathbf{X} \rightarrow \begin{bmatrix} \mathbf{Y}^1 \\ \mathbf{Y}^2 \\ \mathbf{X}^3 \end{bmatrix} \quad (\text{Eq. 6-20})$$

假定使用傳統籬柵解碼時，接收端解出的訊號樣式，不存在 {Eq. 6-15 ~ Eq. 6-20} 的型態。故 DSTTC 具備 full diversity，對於任何 STC 中，相異的 \mathbf{X} 與 \mathbf{Y} ，Eq. 6-13 與 Eq. 6-14 所形成的差異矩陣，其 rank 為 3，也就是指傳統籬柵解碼的 diversity 為 3。但矩陣 (Eq. 6-13) 與矩陣集合 (Eq. 6-15 ~ Eq. 6-20) 所形成的差異矩陣卻不是 full rank。當使用混合籬柵解碼時，因為 (Eq. 6-15 ~ Eq. 6-20) 的矩陣型態，都有可能發生，故我們必須詳加分析。當傳送端欲傳 \mathbf{X} 資訊時，混合籬柵解碼經過多數決以後，因為多數決混合籬柵解碼會將 {Eq. 6-15 ~ Eq. 6-17} 的矩陣，視為傳送端傳遞 \mathbf{X} 資訊，所以 Eq. 6-13 誤會成 {Eq. 6-15 ~ Eq. 6-17} 中的某個矩陣，其 pair-wise 錯誤機率皆不被包含在 Eq. 6-12 中；然而，因為多數決混合籬柵解碼會將 {Eq. 6-14, Eq. 6-18 ~ Eq. 6-20} 的矩陣，視為傳送端傳遞 \mathbf{Y} 資訊；Eq. 6-13 誤會成 {Eq. 6-14, Eq. 6-18 ~ Eq. 6-20} 中的某個矩陣，仍包含在 Eq. 6-12 中；又 Eq. 6-13 誤會成 Eq. 6-14 的錯誤率 diversity 為 3，Eq. 6-13 誤會成 {Eq. 6-18 ~ Eq. 6-20} 中某個矩陣的錯誤

率 diversity 為 2，所以多數決混合籬柵解碼應用於中繼端群編碼器狀態一致情況下，diversity 為 2(由 diversity 小的支配錯誤率斜率)。

圖 6-18，使用混合籬柵解碼，分散式籬柵碼的衍生係數可參考表一，在中繼端群編碼器狀態一致時，錯誤率曲線的 diversity 為 2，較傳統解碼方式(diversity 為 3)表現差。不過混合籬柵解碼方式，無 AWGN 時，不管在中繼端群編碼器狀態是否一致的情況下，皆能正確解碼，這是傳統籬柵解碼方式不具有的特性。

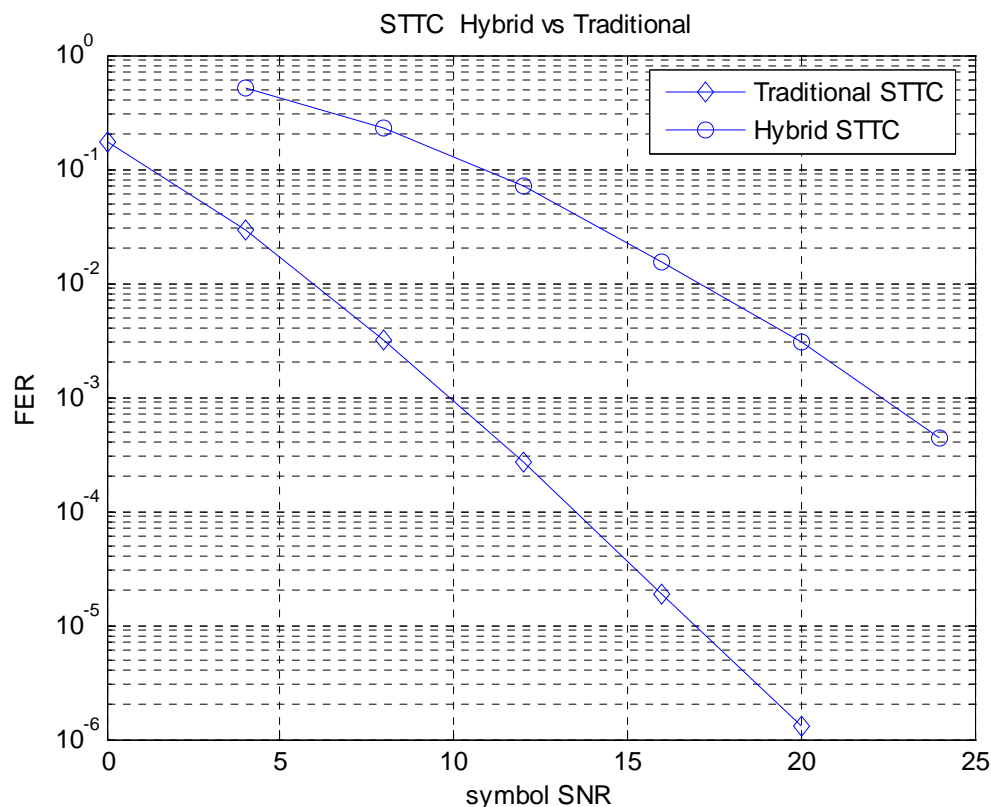


圖 6-18 中繼端狀態一致下，傳統解碼與混和解碼錯誤率比較圖

6.5.7 多數決混合籬柵解碼 diversity 特性總結

受干擾中繼端的輸入訊號與其它中繼端不同時，產生中繼端群編碼器狀態不一致問題；使用多數決混合籬柵解碼，能夠有效偵測傳送端的訊號，其解碼錯誤率的 diversity 為 1；然而，使用傳統籬柵解碼時，無法有效偵測傳送訊號，所以解碼錯誤率的 diversity 為 0。當中繼端群的輸入訊號一致時，使用多數決混合籬柵解碼，也能夠有效偵測傳送端訊號，其解碼錯誤率的 diversity 為 2；然而，使用傳統籬柵解碼時，其解碼錯誤率的 diversity 為 3。

diversity	Traditional decoding	Hybrid trellis decoding
Consistency	3	2
Inconsistency	0	1

觀察表 6-2，明顯發現混合籬柵解碼方式犧牲了一個 diversity，用來更正一個中繼端的解碼錯誤，使得中繼端群編碼器狀態不一致時，也能正確解碼，這是傳統籬柵空時解碼

不同的地方。

6.5.8 應用 M-Algorithm 於混合式籬柵解碼

為了偵測受干擾中繼端的訊號，我們使用混合式籬柵解碼；比起傳統的籬柵空時碼，混合式籬柵解碼多了延伸狀態，以幫助了解受干擾中繼端的編碼器輸入訊號。這些多出來的延伸狀態，增加解碼的複雜度。為了降低解碼複雜度，使用 M-algorithm 於其中，簡化維特比演算法的複雜度。在每一個 trellis level 只留下 M 個 survivor，留下的原則，是根據 path metric 所累積的距離大小做"sorting"，留下最小的 M 個。如此一來，在計算複雜度可以降低；此外，假如暫存器個數愈多的話，可以刪掉的 survivor 愈多，對於高 state 個數的編碼方式，更能有效降低複雜度。注意，當 $M=2^{\text{暫存器個數}}$ ，複雜度跟傳統維特比解碼相同。

圖 6-19 為在中繼端群編碼器狀態不一致時，M-Algorithm 應用於不具多數決混合式籬柵解碼效能比較圖；分散式籬柵編碼衍生係數請參考表 6-2，每個中繼端使用 4-states QPSK 籬柵空時編碼器；接收端使用"不具備多數決檢查混合式籬柵解碼"，考慮系統中，至多一個受干擾中繼端，因此 Hybrid 籬柵中，正常中繼端與受干擾中繼端的組合，共有 16 個 states，由於接收端不知何者中繼端為受干擾中繼端，所以混合籬柵碼必須將 states 個數增為原來的 3 倍，解碼端共計 48 個 states。由模擬結果觀察，我們可以發現，當 $M=4$ 時，效能降低的幅度不會很多，顯示 M-Algorithm 能有效降低複雜度並維持解碼效能。

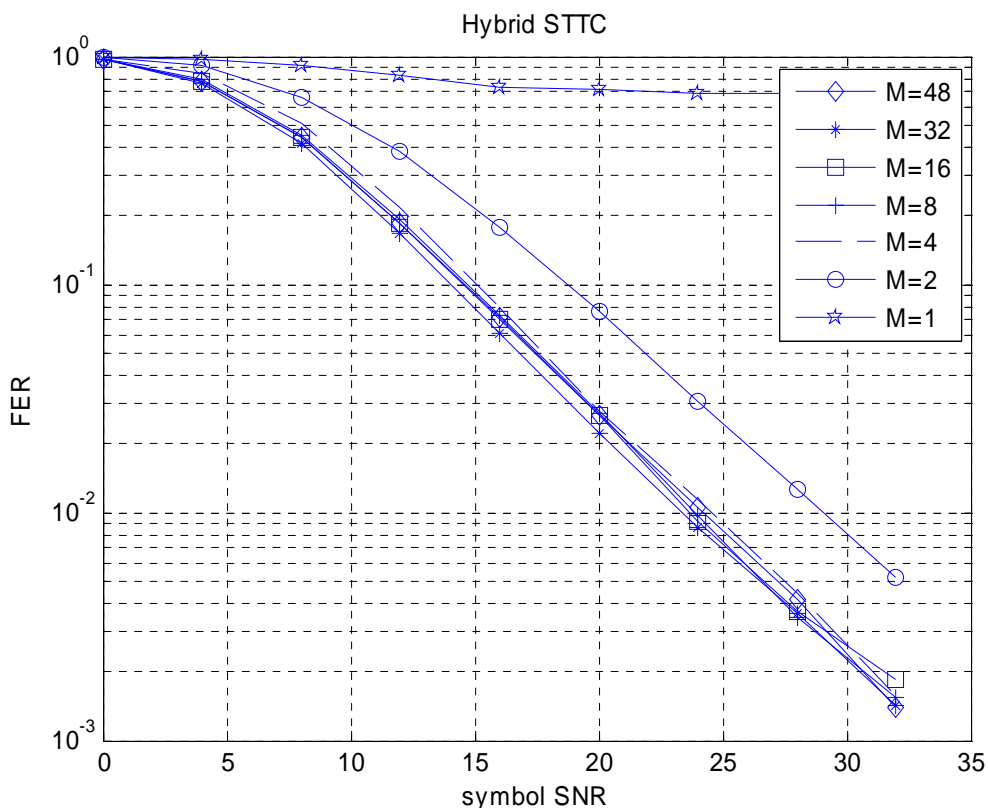


圖 6-19 M-Algorithm 應用於混合式籬柵空時碼來降低解碼複雜度

表 6-2 4-states 籬柵空時碼

籬柵空時碼	Generator sequence
Code C	$g^1=[(022),(122)]$ $g^2=[(233),(202)]$



第七章 結論

籬柵空時碼能夠有效降低錯誤率，根據前人所提出的設計準則，我們提出有效率的碼搜尋演算法，使得尋找高複雜度的籬柵空時碼時，降低搜尋的時間。碼搜尋的方法，最主要是利用籬柵特性，不再使用費時的全域搜尋方法，應用遞迴特性與 Viterbi 演算法，加快搜尋的收斂速度。要將籬柵空時碼運用於任何系統中，搜尋效能好的籬柵空時碼是第一步，期望籬柵空時碼能發揮系統架構的最大極限，如 full diversity、high coding gain。

空時碼應用於分散式系統，是當今流行的話題。電路錯誤、合作式通訊中受干擾中繼端問題等，使得傳統籬柵空時碼的接收方法運作失敗，本篇論文提出兩種解決方式：分離式解碼與混合式籬柵解碼。分離式解碼方式不失為此問題的解決方法，但不屬於 ML 解法，在錯誤率表現上不佳。混合式籬柵解碼在錯誤率表現上優於分離式解碼，雖然複雜度高，但可以由傳統簡化的演算法(Slide Window、M-Algorithm、Adaptive Viterbi...)來降低複雜度，使得實作更為務實。由傳統空時碼分析封包錯誤率的方式，並從多重天線分集與多工的消長特性來驗證，我們證明遇到中繼端群編碼器狀態不一致問題時，接收端使用 ML 解法，的錯誤率表現的 diversity 接近 1。傳統空時碼多強調的傳送分集(transmit diversity)，在此因為電路錯誤、收干擾中繼端問題，傳送分集的特性(full diversity)消失。混合籬柵碼的編碼增益，可以從各個天線所屬的編碼增益來分析，增加各自天線的所屬編碼增益，能有效增加混合籬柵解碼方式的編碼增益。此外，接收端經由多數決檢查，產生的選擇增益，對於僅要判斷傳送端訊號是否正確而言，是能降低錯誤率的方式之一。在中繼端群編碼器狀態一致時，使用具備多數決混合解碼方式的錯誤率，diversity 甚至能達到 2。

經由電腦模擬驗證，關於搜尋碼的部分，結合遞迴特性與 Viterbi 演算法確實能夠使搜尋時間降低許多；關於分散式系統，中繼端群狀態不一致問題，所提出的兩種解碼方式，確實可以解決我們所遇到的問題，不會像傳統籬柵空時碼解碼時，完全解不出來，提供 DF 合作式通訊中，遇到中繼端群編碼器狀態不一致時，籬柵空時碼的實現方式。

未來展望：使用混和籬柵編碼時，其解碼複雜度隨著狀態與 branch 的個數變多而增加，我們可以試著朝簡化解法複雜度的方向前進，使之在實際運算上能夠很親近實做。

參考文獻

- [1] G.J. Foschini, “**Layered space-time architectures for wireless communications in a fading environment when using multi- element antennas,**” *Bell-Labs Tech. Journal*, vol. 1, no. 2, pp. 41-59, 1996.
- [2] V. Tarokh, N. Seshadri, and A. R. Calderbank, “**Space-time codes for high data rate wireless communication: Performance criterion and code construction,**” *IEEE Trans. Inform. Theory*, Vol. 44, pp. 744~765, Mar.1998.
- [3] J. Nicholas Laneman and Gregory W. Wornell, “**Distributed Space-Time-Coded Protocols for Exploiting Cooperative Diversity in Wireless Networks,**” *IEEE Trans. on Inform. Theory*, Vol. 49, no. 10, pp2415~2425, Oct. 2003
- [4] Jinhong Yuan, Zhuo Chen, Yonghui Li and Li Chu, “**Distributed Space-Time Trellis Codes for a Cooperative System,**” *IEEE Trans. on wireless Comm*, Vol. 8, no. 10, pp 4897~4905, October 2009.
- [5] Stephan B aro, Gerhard Bauch, and Axel Hansmann, “**Improved Codes for Space-Time Trellis-Coded Modulation**”, *IEEE comm letters*, Vol. 4, no. 1, pp.20-22 January 2000
- [6] Jinhong Yuan, “**Adaptive Transmit Antenna Selection with Pragmatic Space-Time Trellis Codes,**” *IEEE Trans. on wireless Comm.*, Vol. 5, no. 7, pp1706~1715, July 2006.
- [7] Rick S. Blum, “**Some Analytical Tools for the Design of Space-Time Convolutional Codes,**” *IEEE Trans. on Comm.*, Vol. 50, no. 10, pp1593-1599 Oct 2002
- [8] David Tse, “**Fundamentals of Wireless Communication,**” *Cambridge*, 2005
- [9] Branka Vucetic, Jinhong Yuan, “**Space time coding,**” *WILEY*, 2003.
- [10] L. Zheng and D. N. C. Tse, “**Diversity and Multiplexing: A fundamental tradeoff in multiple-antenna channels,**” *IEEE Trans. Inform. Theory*, Vol. 49, no. 5, pp. 1073–1096, May 2003.
- [11] S. A. Zadeh and M. R. Soleymani, “**An adaptive M-algorithm Convolutional decoder,**” *Proc. IEEE Veh. Tech. Conf* 05, pp. 2177-2181, Sep. 2005.
- [12] Junfeng Shi, Zhimin Zeng, “**SER Performance Analysis for jointing Network Coding and Alamout coding in DF Relay System**”.
- [13] S.Lin and D.J.Costello, Jr., “**Error Control Coding, 2nd ed,**” *Pearson Prentice Hall*, 2004.
- [14] Jan Mietzner, Robert Schober, Lutz Lampe, Wolfgang H. Gerstacker and Peter A. Hoeher “**Multiple-Antenna Techniques for Wireless Communications – A Comprehensive Literature Survey,**” *IEEE Comm. surveys & tutorials*, Vol. 11, no. 2, pp87~105, Second Quarter 2009.
- [15] Tatsuki FUKUDA, Shingo OTSU, Yuta TOKUNAGA and Hua-An ZHAO, “**A Realization of Determinant Criterion for STTC Design,**” *ITC-CSCC*, 2008
- [16] J. W. Craig, “**A new, simple and exact result for calculating the probability of error for two-dimensional signal constellations ,**”*In Proc. of MilCom*, pp.571- 571, 1991